On Platforms and Algorithms For Human-Centric Sensing

ON PLATFORMS AND ALGORITHMS FOR HUMAN-CENTRIC SENSING

BY

ALA SHAABANA, M.Sc., (Computer Science)
University of Windsor, Windsor, Canada

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF COMPUTING & SOFTWARE
AND THE SCHOOL OF GRADUATE STUDIES
OF MCMASTER UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2017) McMaster University
(Computing & Software) Hamilton, Ontario, Canada

TITLE:	On Platforms and Algorithms For Human-Centric Sensing
AUTHOR:	Ala Shaabana M.Sc., (Computer Science) University of Windsor, Windsor, Canada
SUPERVISOR:	Dr. Rong Zheng
NUMBER OF PAGES:	xvi, 156

To my parents

Abstract

The decreasing cost of chip manufacturing has greatly increased their distribution and availability such that sensors have become embedded in virtually all physical objects and are able to send and receive data – giving rise to the Internet of Things (IoT). These embedded sensors are typically endowed with intelligent algorithms to transform information into real-time actionable insights. Recently, humans have taken on a larger role in the information-to-action path with the emergence of human-centric sensing. This has made it possible to observe various processes and infer information in complex personal and social spaces that may not be possible to obtain otherwise. However, a caveat of human-centric sensing is the high cost associated with high precision systems.

In this dissertation, we present two low cost and high performing end-to-end solutions for human-centric sensing of physiological phenomena. Additionally, we present a post-hoc data-driven sensor synchronization framework that exploits independent, omni-present information in the data to synchronize multiple sensors. We first propose XTREMIS – a low-cost and portable ECG/EMG/EEG platform with a small form factor that has a sample rate comparable to research-grade EMG machines. We evaluate XTREMIS on a signal level as well as utilize it in tandem with a Gaussian

Mixture Hidden Markov Model to detect finger movements in a rapid, fine-grained activity – typing on a keyboard. Experiments show that not only does XTREMIS functionally outperforms current wearable technologies, its signal quality is high enough to achieve classification accuracy similar to research-grade EMG machines, making it a suitable platform for further research. We then present SiCILIA – a platform that extracts physical and personal variables of a user's thermal environment to infer their clothing insulation. An individual's thermal sensation is directly correlated with the amount of clothing they are wearing. Indeed, a person's thermal comfort is crucial to their productivity and physical wellness, and is directly correlated with morale [18]. Therefore it becomes important to be aware of actions such as adding or removing clothing as they are indicators of current thermal sensation. The proposed inference algorithm builds upon theories of body heat transfer, and is corroborated by empirical data. SiCILIA was tested in a vehicle with a passenger-controlled HVAC system. Experimental results show that the algorithm is capable of accurately predicting an occupant's thermal insulation with a low mean prediction error. In the third part of the thesis we present CRONOS – a sensor data synchronization framework that takes advantage of events observed by two or more sensors to synchronize their internal clocks using only their data streams. Experimental results on pairwise and multi-sensor synchronization show a significant drift improvement for total drift and a very low mean absolute synchronization error for multi-sensor synchronization.

Acknowledgements

Many thanks to Rong Zheng for her insights, enthusiasm, and tutelage throughout my Ph.D. work. I hope that one day I will similarly be able to inspire and mentor people. I also wish to thank past and present members and visitors to the Wireless Systems Research (WiSeR) lab as well as my committee: Dr. Douglas Stebila and Dr. Martin von Mohrenschildt for their insights and advice. Finally I wish to thank Nazanin for her unwavering support, encouragement, and many coffee dates below "the bridge".

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In the broadest definition, a sensor can be defined as an electronic device whose purpose is to detect events or changes in its environment and either report the information or store it for later use. The widespread distribution and availability of Micro Electro-Mechanical Systems (MEMS) is transforming the physical world into a ubiquitous computing platform [110] – giving rise to the Internet of Things (IoT) and human-centric sensing. Human involvement is particularly useful in sensing various processes in complex personal, social, and urban spaces where traditional embedded sensor networks tend to suffer from gaps in spatiotemporal coverage, limitations in making complex inferences, and inability to adapt to dynamic and cluttered spaces [97]. By taking advantage of users who already live and work in these spaces, as well as their adaptability and intelligence, human-centric sensing makes it feasible to get information that otherwise is not possible. While human-centric sensing systems are quite diverse, one can classify them in terms of the extent and role of human participation, which falls under one or more of the following categories [97]:

1. **Humans as targets of sensing**: The advent of more pervasively deployed sensor technologies have resulted in an increased interest in applications for sensing of human activities, behaviors, and patterns at scales ranging from individuals (often oneself) to larger groups and communities.

2. **Humans as sensors**: A second role for humans is to participate in community sensing campaigns, either via explicit recruitment or implicitly by downloading a sensing application.

3. **Humans as data sources**: Humans regularly act as data sources themselves, acquiring and disseminating information on their own, without the aid of sensing devices.

In this dissertation, we construct low cost, high performing end-to-end platforms to investigate human physiological phenomena under these categories. The basis of this study is that systems that infer physiological aspects of the human body such as action potential and heat are rather expensive and bulky. Further, at the time of writing this dissertation there does not yet exist a truly low cost, portable, and high performing counter-part. The objective of this dissertation is therefore to design, create, and evaluate low cost and portable end-to-end systems in which humans are either targets of sensing, or are treated as sensors or data sources to infer personal information such as clothing levels and finger utilization through the analysis of physiological signals with high efficiency and performance.

1.1 Contributions

The main contribution of this dissertation is that it provides low-cost and high performing tools for human-centric sensing to collect physiological data and make predictions about user physical states or actions. The proposed systems fall into two categories: contact and contactless systems. Additionally, we propose a framework to synchronize the data streams from multiple sensors post-hoc.

We first present a contact system – codenamed XTREMIS – a low-cost and portable Electrocardiograph (ECG)/Electromyograph (EMG)/Electroencephalograph (EEG) platform with a small form factor that has a sample rate comparable to research-grade EMG machines. We utilize XTREMIS to collect EMG data in order to infer finger movement in a keyboard typing activity. However, due to the fast moving nature of fingers, it is difficult to predict which finger moved for a given key press using discriminant classifiers. We thus present a Gaussian Mixture Hidden Markov Model (GM-HMM) to predict the finger movements by taking advantage of the finger movement patterns present in the training data. We compare the performance of the GM-HMM with other classification algorithms such as Support Vector Machines and k-Nearest-Neighbor. Additionally, we analyze the factors affecting fine-grained gesture recognition such as sampling rate and electrode placement. Finally, we compare XTREMIS' performance to the Myo armband[41] – a commercial off-the-shelf EMG sensor, and the BioSemi ActiveTwo [10] – a research-grade EMG system.

We then present a contactless system – codenamed SiCILIA – a platform that extracts physical and personal variables of a user's thermal environment to infer the amount of clothing insulation without human intervention. A human's thermal environment has a myriad of factors that affect their thermal sensation (i.e. cold, hot, warm, etc.) such as the ambient temperature, the clothing temperature, and the clothing emissivity. Additionally, a person's thermal sensation is important as it is correlated with their productivity, alertness, and can lead to illnesses such as the Sick-building syndrome [77]. Since clothing is the most direct factor to human thermal sensation, we developed a hardware platform as well as a real-time clothing detection algorithm based on prior works in textile manufacturing to infer the level of clothing insulation worn by the user.

Finally, we present CRONOS – a post-hoc, data-driven framework for sensor data synchronization in ubiquitous environments. Traditional synchronization methods aim to synchronize the clocks on sensors and generally rely on wireless communication protocols (e.g. RBS [16]) or modifying hardware (e.g. internal real-time clocks (RTC)). Re-engineering entire systems to facilitate synchronization can quickly become costly as the number and types of devices grow. Further, adding more hardware to existing systems (i.e. upgrading with an RTC or a high quality oscillator) is likely to increase cost and power consumption. CRONOS takes advantage of independent, omnipresent events observed by two or more sensors to synchronize their internal clocks offline using only their data streams without recalibrating or adjusting their internal clocks.

1.2 Research Challenges and Objectives

As of late, large companies have been investing into maximizing worker comfort by creating "health and wellness" divisions to analyze factors such as ergonomics, noise, and personal spaces to create comfortable, personalized environments for their workers. This is due to the fact that workers will only be as productive as they are comfortable [18]. However, the analysis performed by these teams is rather cumbersome (e.g. type on a keyboard repeatedly for ergonomic analysis, sit while body measurements are being taken for posture analysis, etc.). The objective of this work therefore is to develop systems that are capable of analyzing body emission information for potential use by industry professionals to maximize and analyze worker comfort. However, systems that can be used for this type of analysis are rather expensive as they are highly precise. Therefore the first challenge is to develop a cheaper system that is capable of performing the same tasks with similar accuracy.

The second challenge is in the algorithms to extract and analyze the human body emission data. In the case of XTREMIS, finger movement recognition in a naturalistic typing setting is a difficult problem due to the high amount of noise between finger movements – making it difficult to differentiate between key presses and releases. Similarly in SiCILIA, the problem of quantifying clothing insulation is difficult due to the factors surrounding textiles and the heat exchange processes surrounding the human body (convection, radiation, and conduction). Finally in the case of CRONOS, there are multiple challenges surrounding data-driven sensor synchronization such as synchronizing between sensors with different modalities (e.g. synchronizing SiCILIA and XTREMIS), finding opportunities to synchronize, and synchronizing regardless of phase differences between the data streams.

Combining SiCILIA and XTREMIS, we present solutions capable of extracting information pertinent to an office worker's current physical condition – such as their level of clothing at different indoor temperatures and their typing habits – that are low-cost but highly accurate that can be used by professionals for comfort analysis. Additionally, CRONOS enables us to use the data from sensors that observe different phenomena (e.g. XTREMIS observing finger movements, SiCILIA observing clothing insulation changes) to synchronize between them regardless of their sensing modality.

1.3 Organization

Before constructing any human-centric sensing platform, it is important to first understand the underlying biology of the platform's target. Indeed, building an end-to-end platform without knowledge of the underlying biology will result in a longer development time, a bug-ridden system, and inaccurate data. Chapter 2 introduces the two key biological aspects wof XTREMIS and SiCILIA: heat transfer and muscular physiology, respectively.

Chapter 3 presents XTREMIS present the hardware design and software architecture of XTREMIS. We also identify ballistic gestures: a new type of hand gesture that involves spontaneous propulsion of the limbs. We then present a GM-HMM to identify it and compare its performance on:

1. **Myo armband**: A commercial off-the-shelf EMG armband [41].
2. **BioSemi ActiveTwo**: A research-grade EMG/EEG machine [10].
3. **XTREMIS**: The proposed biomedical platform.

The results of this chapter have been published at the 2^{nd} IEEE/ACM International Conference on Connected Health (CHASE) 2017 [88], and submitted for publication in IEEE Transactions on Embedded Computing Systems [85]. Additionally the code, hardware schematics, and execution manual have all been made available in the public domain on Github for those interested in replicating XTREMIS or studying it [93].

Chapter 4 changes the focus to detecting electromagnetic waves – specifically infrared (IR) – emitted from the human body. Utilizing contactless technologies such as IR sensors and ultrasonic range finders, SiCILIA tracks its user and measures their clothing insulation. The formulaic approach of the tracking and clothing inference algorithms, and the results of the experiments are then presented. SiCILIA's demonstration earned a best demo award at the ACM Cyber-Physical Systems (CPS) week [91]. Additionally, the results of this chapter were published in 2015 at IEEE GLOBECOM [90] and they have been accepted and are currently in press in ACM Transactions on Sensor Networks (TOSN) [92]. Similarly to XTREMIS, the code has been made available in the public domain on Github for those interested in the research [84].

Chapter 5 changes the focus once again and presents CRONOS – an entirely software-based approach to sensor synchronization. For the pairwise sensor case, the sensor time skew and offset are first detected and corrected with a combination of normalized cross correlation and linear regression. For the multi-sensor case, a graph-based approach to multi-sensor synchronization with least squares optimization to synchronize using the most up-to-date synchronization data is presented. The results of this chapter have been submitted to the ACM Proceedings of the 17th International Conference on Information Processing in Sensor Networks (IPSN) [87]. The python notebooks for CRONOS are available on Github [86]. Finally, chapter 6 concludes this dissertation and discusses the future works.

Chapter 2

Background

Measurements of human biological processes tend to be noisy, and sufficient knowledge of the process that is under measurement is crucial. We present some biological context regarding the underlying muscular physiology on which the finger movement recognition using XTREMIS is based. We then present the underlying heat transfer mechanisms that SiCILIA exploits to create its predictions.

2.1 Muscular Physiology

Instead of being controlled with interior muscles, human fingers function in a pulley system powered by the forearm. In fact, there are 20 muscles in the forearm that control fingers and movement. They can be divided into 2 types: extrinsic and intrinsic muscles as shown in Figure 2.1. The extrinsic muscles are the long flexors and extensors. The flexors are located on the underside of the arm, and allow for the bending of the fingers [98]. The thumb has one long flexor and one short flexor, as well as other muscles to make grasping possible. The extensors, on the other hand, are located on the back of the forearm and they help to straighten fingers out (i.e. finger extensions).

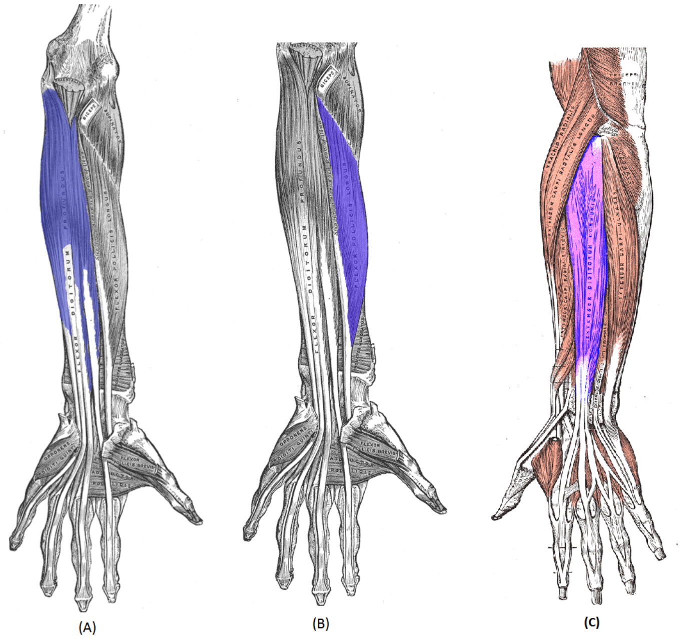


Figure 2.1: (a) Flexor Digitorum Profundus (b) Flexor Pollicis Longus muscles (c) Extensor Digitorum Communis [28]

The Flexor Digitorum Profundus' primary functions are the flexing of the wrist, the metacarpophalangeal joints (joints between the bones and phalanges of the fingers), and the interphalangeal joints (hinge joints between the phalanges of the hand). In other words, it helps in flexing the medial four digits of the hand (the index, middle, ring, and pinky fingers). Whereas the Flexor Pollicis Longus muscle serves to primarily flex the thumb. Meanwhile, the Extensor Digitorum Communis muscle allows for the extension of the medial four digits of the hand. Located on the back of the forearm, this muscle is in a constant state of contraction when typing due to the posture of human beings when typing.

2.1.1 Electromyography

In any cell, there exists an electrical potential difference between the inside of the cell and the surrounding extracellular fluid called the membrane potential of the cell. A phenomenon referred to as an Action Potential (AP) occurs when the membrane potential of a specific axon (i.e. a nerve cell) rapidly rises and falls, causing adjacent locations to similarly depolarize [36]. APs occur in several types of animal cells – called excitable cells – which include muscle cells. Specifically, AP produces the contraction required for the movement of muscles, therefore AP levels are directly correlated with the intensity and nature of muscle movements.

Electromyography is an electro-diagnostic tool utilized for the evaluation and recording of the electrical activity produced by skeletal muscles [47]. Specifically, an Electromyograph detects the aggregate AP generated by muscle cells when they are electrically or neurologically activated. The signals are analyzed to detect medical abnormalities, activation levels, recruitment order, or to analyze the biomechanics of an animal or a human. Figure 2.2 shows a simplified action potential signal that occurs when a muscle is contracted or relaxed. The depolarization and repolarization phases occur first as the membrane potential of the muscle cells rises and falls.

2.2 Biological Heat Transfer Mechanisms

Humans are homeothermic, meaning they must maintain their body temperatures within a narrow range in varying environmental conditions. Typically, body temperatures range between 36-37 °C [40]. However, there is a diverse range of thermal sensations from one human to another that is caused by the variance in their resting

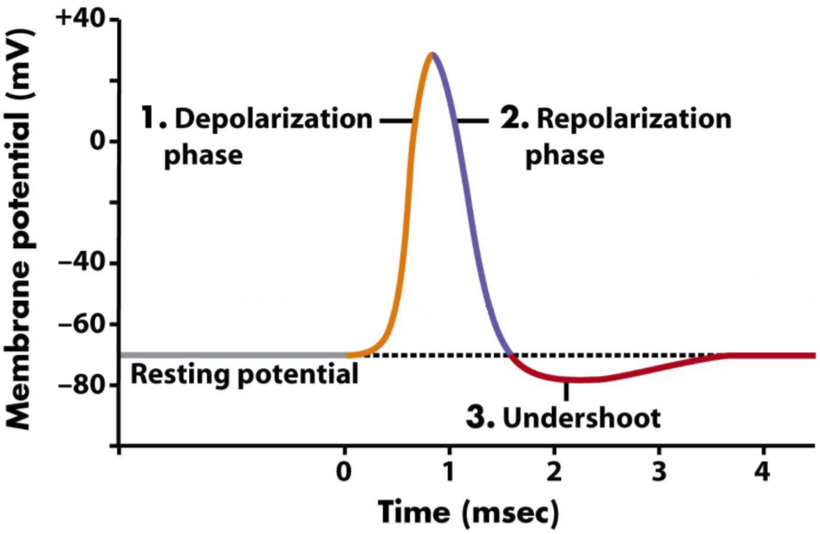


Figure 2.2: The shape of an action potential signal during resting, polarization, depolarization, and the undershoot (i.e. refractory) period.

core body temperature. This variation means that humans tend to have different tolerances to working at certain temperatures. Further, there are two recognized sources of heat load on each person: the first is environmental, which may be positive or negative (i.e. there may be a heat gain or loss from the body due to the surrounding environment). The second is metabolic, which is generated by muscular activity (or work). Hence, a human's thermal sensation is determined by the sum total of the effects of these processes. Meanwhile their physiological make-up determines their preferences and level of comfort, given these processes.

As illustrated in Figure 2.3, the main processes of heat exchange between the human body and its environment are [8]:

- **Convection:** The difference in temperature between the air and the skin temperature along with the rate of air movement over the skin decides the rate of convection. Convection rate characterizes the rate of convective exchange

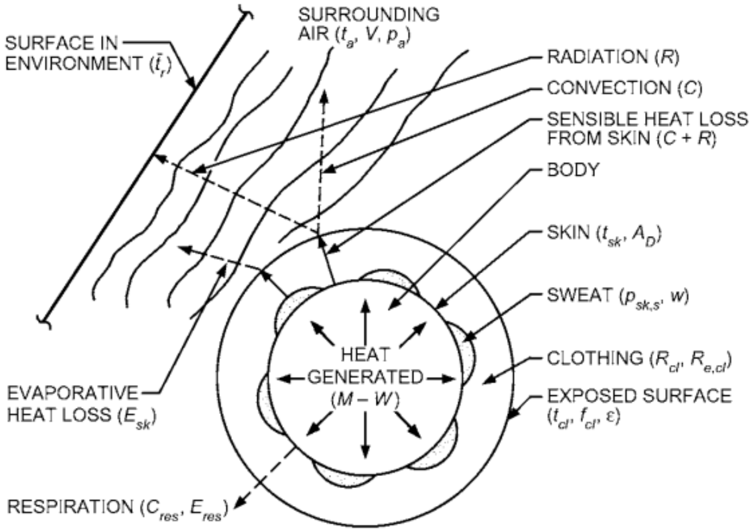


Figure 2.3: Thermal Interaction of an occupant's body and its environment [31]

between human skin and the ambient air in close proximity to the skin.

- **Radiation:** The surface of the human body constantly emits heat in the form of electromagnetic waves. The rate of emission is determined by the absolute temperature of the radiating surface. Therefore, net heat is lost if the surface of the body is warmer than the average of the various surfaces in the environment, with the rate being directly dependent on the temperature difference. This form of heat transfer does not require direct contact with the warmer object.

- **Conduction:** In the case of conduction, the heat moves down its thermal gradient from the warmer to the cooler object. After it has lost some of its heat, the warmer molecule slows down and the cooler molecules move faster after having gained heat. The heat transfer between the two objects continues until their temperatures have reached equilibrium (that is, equalized).

- **Evaporation:** To cool the body, water evaporates from the surface of the skin.

> Additionally, the heat required to transform it from a liquid to a gas is dissipated from the skin.

Therefore, convection, radiation, conduction, and *sometimes* evaporation are the main processes that affect persons at rest in an indoor environment such as an office.

Chapter 3

XTREMIS: A Portable Biomedical Sensing Platform

Electromyography (EMG) is an electro-diagnostic tool used to analyze muscle response or electrical activity in response to a nerve's stimulation of the muscle. This is done by detecting a small bioelectric pulse called the Action Potential (AP) that is generated when muscles contract and relax. By analyzing the AP signal, it becomes possible to analyze medical abnormalities, activation levels, muscle recruitment order, or biomechanics [78]. Currently there exist two types of EMG: intramuscular EMG and surface EMG (sEMG). Intramuscular EMG requires an electrode to be inserted into the muscle that is being measured. This is an invasive process and can be painful to the person being measured, however the signal is typically clear with a low amount of noise. On the other hand, surface EMG electrodes are placed on the skin, making it non-invasive and pain-free. However, the signal is noisy and requires proper filtering before being processed.

There has been a myriad of research on the use of sEMG to to drive an actuation based on human biosignals in the past couple of decades. Specifically, sEMG signals are processed and input into classifiers to create functions ranging from hardware control to gesture recognition. Indeed, one of the earliest concepts of using EMG for robotics control was proposed by Farry *et al.* in 1996, in which they proposed converting EMG signals into commands for NASA/Johnson Space Center's sixteen degree-of-freedom Utah/MIT Dexterous Hand for two grasping (key and chuck) options and three thumb motions (abduction, extension, and flexion) [20]. As of late, EMG research has been more focused on the classification of intricate tasks. This naturally led researchers to finger movement classification. In 2013, Chen *et al.* proposed a pattern recognition system to perform automatic classification on multiple finger movements, specifically Chinese sign language gestures for numbers ranging from 0 to 9 [12]. Moreover, they investigated the effects of different feature and classifier combinations in offline recognition, and have taken a further step by implementing a real-time recognition system with above 90% accuracies for all subjects.

However, there exists a class of fine-grained gestures that involve repetitive, spontaneous propulsions of the limbs in activities such as playing instruments or typing which we define as "ballistic gestures". We evaluated ballistic gesture recognition performance using a Myo gesture recognition armband [41] and compared it against a Biosemi ActiveTwo research-grade EMG machine [10] and found that while the BioSemi ActiveTwo achieved a high classification accuracy, the Myo armband suffers from two significant setbacks that prevented it from achieving a high classification performance [89]:

1. **Rigid electrode placement**: The electrodes can only be placed in one specific way on pre-specified muscles fitting the primary application of the device. For

example, Myo's primary purpose is gesture recognition, and hence it can only be placed on forearm muscles near the elbow [41].

2. **Limited sampling rate**: Due to power constraints, maximum device sampling rates are typically at approximately 200 Hz. Significant information in the signal may be lost as this does not satisfy the Nyquist-Shannon sampling theorem as typical muscle motor unit action potential is typically between 10Hz and 500Hz [60].

Indeed, although the BioSemi ActiveTwo performs rather well, the high costs and complexity of such research-grade EMG machines prevent developers and scientists without an electronics engineering background from utilizing them for their research needs, instead resorting to off-the-shelf wearable devices like Myo. Thus, we present XTREMIS: a low-cost, portable, and powerful hardware and software solution to EMG data collection. XTREMIS is an EMG data collection device with an adjustable biomedical instrumentation chip and flexible electrode placement. In order to evaluate the validity and performance of XTREMIS, two types of experiments were performed: finger movement recognition and a signal-level analysis and similarity measure. The performance of XTREMIS is also compared against the BioSemi ActiveTwo and the Myo armband. Thus, our contribution in this chapter is four-fold:

1. We present XTREMIS: a hardware platform and software solution for Electrocardiography (ECG/EKG), Electromyography (EMG), and Electroencephalography(EKG) data collection.

2. We propose a Gaussian mixture Hidden Markov model (GM-HMM) to classify ballistic gestures during a typing task.

3. We present a detailed evaluation of the different factors that affect finger movement classification during typing, namely:
    - 3.1. speed of typing.
    - 3.2. placement of electrodes on the skin.
    - 3.3. sampling rates.
4. We evaluate XTREMIS' performance on EMG signals by comparing it to a research-grade EMG machine in two dimensions:
    - 4.1. Perform a signal-level analysis on XTREMIS and compare its performance to a research-grade EMG machine.
    - 4.2. Evaluate the performance of both devices in classifying ballistic gestures using a Gaussian Mixture Hidden Markov Model (GM-HMM).

The rest of this chapter is organized as follows: Section 3.1 discusses gesture recognition and previous works. Section 3.2 discusses the hardware design and software architecture of XTREMIS. Section 3.3 details the GM-HMM applied in recognizing finger movements in the classification performance evaluation of XTREMIS. Section 3.4 highlights the evaluation results of the classifier performance using XTREMIS v.s. a research-grade EMG machine and presents a signal comparison between the two devices. Finally, Section 3.6 presents the conclusion and future works, respectively.

3.1 Related Work

Gesture recognition is a problem that has been tackled using many approaches by researchers since the 1990s. Due to the variety of possible gestures performed by hands, gestures can be identified in two types:

1. *General gestures*: gestures involving the movement of the entire hand or flexing of the fingers such as sign language.
2. *Ballistic gestures*: gestures involving spontaneous propulsion of the limbs in a continuous manner, such as typing.

Gesture recognition literature has been approached using a variety of methods: gloves [3], acoustic [64], vision-based [62, 99], and EMG-based[41, 12, 89]. Therefore they can be broken down to two approaches: device and device-free approaches.

3.1.1 Device-free Approaches

Most alternative approaches to using EMG devices for gesture recognition involve using computer vision and image processing. Mantecón et al. proposed a gesture camera-based machine interface system and used compressive sensing to reduce the dimensionality of the feature vector [57], and utilized an SVM for recognition. The system was tested on 11 different hand gestures made by 6 subjects, and achieved accuracies as high as 100% for certain gestures. Meanwhile, Jiang et al. proposed a solution to recognize features made in front of a Microsoft Kinect sensor [46]. Both Red/Green/Blue(RGB) images and depth images were explored. Features extracted included: histogram of oriented gradient features, local binary pattern features, structure features, and 3D-voxel features. A novel deflation-based orthogonal discriminant analysis was then explored for further feature reduction and enhancing the discriminative ability, which allows for a higher flexibility than LDA. The proposed method excels in its higher accuracies and low complexity implementation. However, the issue that remains with camera-based approaches is two fold:

1. *Privacy:* If the application of the approach is a personal one, then users may have privacy concerns with regard to how the recorded media is being utilized. Indeed, some institutions may wish to keep them as training data for more generic classification investigations.

2. *Granularity:* Camera based approaches are a good solution when dealing with coarse gestures or body language recognition. However if the gestures require finger manipulation, then several factors may come into play:

    2.1. *Background of the user:* In optimal lab conditions, backgrounds are static during testing; that is, they do not move or have much detail behind them. In a more day-to-day environment this may not be the case, and may introduce complications.

    2.2. *Movement:* The advances of lens technology today has allowed for high picture resolutions, allowing us to zoom and focus on specific parts of pictures. However, if the gestures are being performed live and the gesturer is moving while gesturing, this adds a spatial dimension as well as an angular dimension to the problem.

Other works focused on using acoustic signals for gesture recognition. Deyle *et al.* proposed Hambone, a wrist- or ankle- mounted wearable gesture recognition device. Hambone uses two small piezoelectric sensors that pick up the sounds generated by the user's hands or feet via bone conduction [14]. The signals are identified using a HMM and are mapped to a set of commands to control an application. More recently, Gupta *et al.* proposed SoundWave, a technique that leverages the speaker and microphone already embedded in most laptops to sense in-air gestures around the device [29]. An inaudible tone is generated that gets frequency-shifted when it reflects off moving objects like the hand. The shift is measured with the microphone to infer various gestures [29].

Additionally, wireless signals have also been utilized for gesture recognition. Pu *et al.* proposed WiSee – a gesture recognition system that leverages Wi-Fi and the Doppler effect to enable sensing and recognition of human gestures in an entire area such as a house or an office. WiSee performed with high accuracy on course movements such as pushing, pulling, kicking, etc. [74] but will not perform well with fine grained gestures such as finger movements due to the resolution of the signal relative to the small Doppler shift created by fingers.

3.1.2 Device-based Approaches

Device-based gesture recognition largely solves the privacy issue as an ocular view of the user is not required, unless they do not wish for their data to be shared. However, granularity still remains an issue as finger-level recognition has not been thoroughly investigated yet at the time of writing this thesis.

Wang *et al.* proposed RF-IDraw, an RFID-based system that allows a user to interact with a desired computing device by gesturing or writing commands in the air [102]. RF-IDraw is finger-mounted and can detect finger movements on a fine level using a few antenna pairs with different spacial separations between them. However, RF-IDraw is single-finger mounted, meaning it can only be placed on one finger and therefore has not been tested for multiple finger movements. It serves well as a human-computer interaction device, but more complex applications require adjustments to the entire system.

Kim *et al.* proposed a 3D hand motion tracking and gesture recognition system using a data glove with three 3-axis accelerometer sensors, one microcontroller and one Bluetooth module. They developed a rule-based algorithm to track simple hand gesture recognition [49] but had significant delays. Asokan *et al.* proposed ARMatron – a wearable gesture recognition glove with on-board potentiometers (one for each finger) to measure finger movement, and an accelerometer and a gyroscope to measure the wrist's rotation and orientation to enable interfacing with electro-mechanical systems and human-computer interactions [3]. Kalpattu *et al.* proposed a glove-based sign language translator using capacitive touch sensors at its core. More specifically, the glove employs touch sensors at the fingertips and the skin between the fingers to collect data [1].

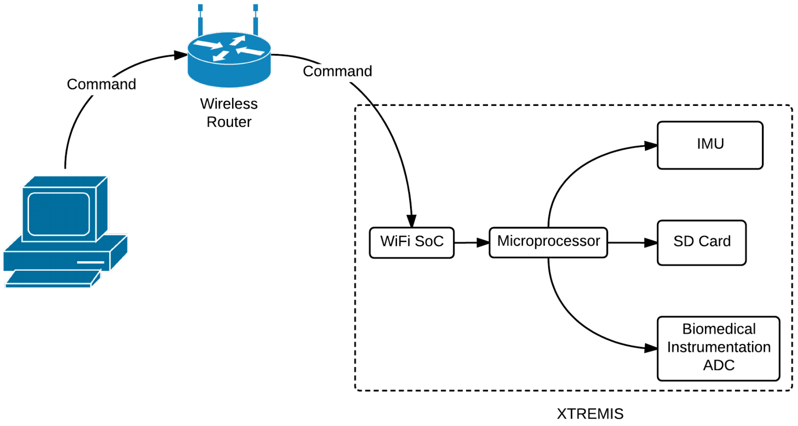
Various EMG solutions to general gesture recognition have been developed using a combination of custom hardware and software. One of the earliest works on the application of EMG on Human Computer Interfaces was proposed in 1998 by Rosenberg in which a graphic input device controlled by the wrist is constructed. The device detects the EMG signal of the forearm muscles used to move the wrist and moves the mouse pointer on a screen accordingly. Rosenberg reports that the pointer performs 14% as well as a regular computer mouse at simple pointing tasks [80]. In 2009 Gopra *et al.* proposed an EMG based control method for an upper-limb motion assisting exoskeleton (SUEFUL-7) with 7 degrees of freedom [27]. SUEFUL-7 takes advantage of the EMG signal amplitudes produced by the upper-arm and forearm muscles to predict the intended movement of the wearer. Experiments have shown that the system is effective in helping physically weak individuals to rotate their shoulders and extend/flex their wrists [27].

Chen *et al.* proposed a pattern recognition system to perform automatic classification on multiple finger movements, specifically Chinese sign language gestures for numbers ranging from 0 to 9 [12]. The proposed hardware system consisted of an instrumentation amplifier and two TelosB motes as an analog-to-digital converter and to wirelessly transmit data. Although their design is compact, each channel requires two electrodes at a time — one reference and one channel electrode. This design fits their implementation with 4 channels [12] but quickly becomes cumbersome when dealing with high electrode counts such as 8 or 16. XTREMIS resolves this issue by using one common reference electrode that works with each channel individually, eliminating the clutter.

Pareschi *et al.* designed an analog-to-information converter based on compressed sensing that acquires biosignals with Nyquist frequency up to 100kHz [71]. Compressed sensing is utilized to reduce the amount of data necessary to represent the signal information content. Further, the proposed system contains signal saturation checking mechanisms to allow users to reconstruct the input signal regardless of the presence of saturation with minimal hardware requirement costs. Experiments on biomedical signals show that the prototype is capable of successfully acquiring signals with high compression factor [71].

3.2 Hardware Design

The design of XTREMIS is inspired by the Open Brain Computer Interface (OpenBCI) open source project [43] – a bluetooth-based ECG/EMG/EEG system. The high-level user flow of XTREMIS is shown in Figure 3.4. Commands are sent from a computer connected to the same wireless network as XTREMIS. They are then received by the WiFi system-on-chip(SoC) and sent to the processor. The processor, in turn, converts the commands to bytes and sets or clears flags on the 6-degree Inertial Measurement Unit (IMU) or biomedical instrumentation ADC (BIADC) as required. During data collection, the incoming digital signal from the BIADC is timestamped by the processor and converted to hexadecimal format. It is then either saved to the SD card, sent to the WiFi SoC for transmission, or both.



oo

Figure 3.4: Information flow of XTREMIS. Commands are sent through WiFi to the WiFi SoC of XTREMIS.

Figure 3.5 is a picture of the XTREMIS board with the highlighted components as shown in Figure 3.4. XTREMIS' compact form factor of 33cm × 55cm makes it suitable for a wide range of applications: from wearable technology to on-the-fly gesture recognition. The architecture of the XTREMIS board is shown in Figure 3.6. For simplicity, we split the circuit into two sub-circuits: data acquisition and data processing.

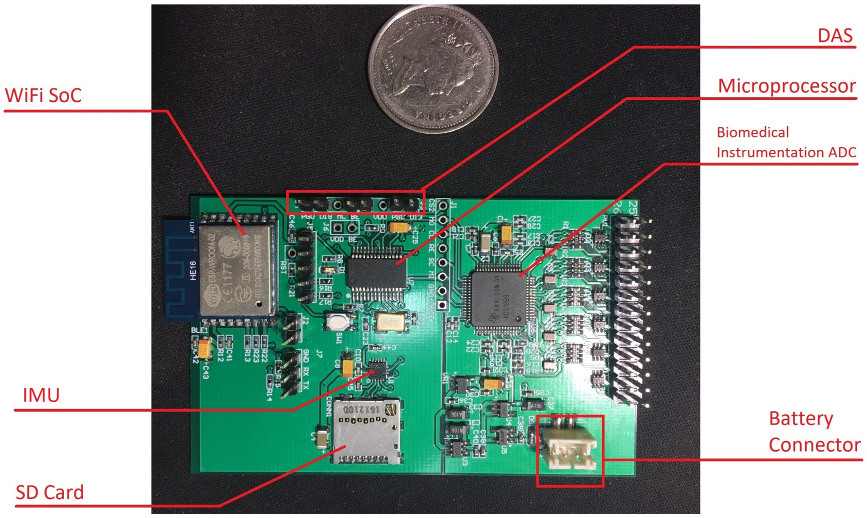


Figure 3.5: XTREMIS Circuit Components.

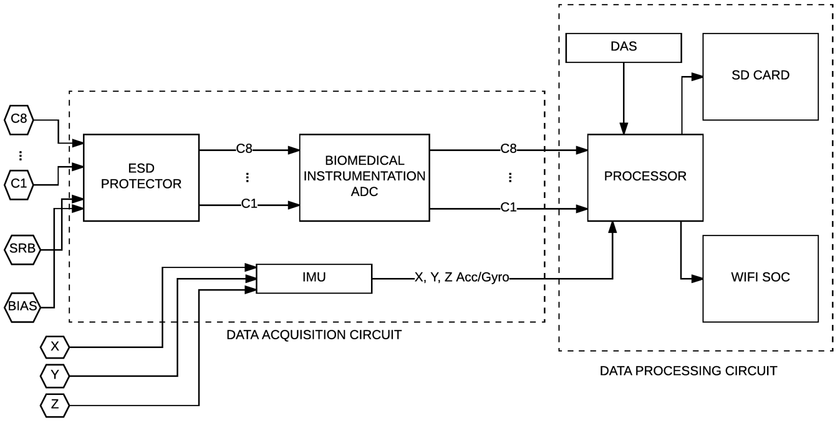


Figure 3.6: High-level view of XTREMIS' architecture.

3.2.1 Data Acquisition Circuit (DAC)

Input to the BIADC is collected from 8 individual channels ($S = C1,...,C8$), a Stimulus, Reference, and Bias (SRB) channel, and a BIAS channel. The SRB channel serves as a common reference to all channels in S against which the impedance on the skin is measured between it and a given channel. As for the BIAS channel (also called Driven Right Leg or DRL), it takes a copy of the 50Hz or 60Hz radiation surrounding the human body and drives it with a scaled, inverted version, effectively canceling it out. It is typically utilized for EEG applications as the signal is weak. The signal for EKG and EMG is stronger and hence it is not required. The BIAS channel input was added to XTREMIS for potential EEG applications in future works.

More specifically, the BIADC is a TI ADS1299 chip that collects the ECG/EMG/EEG data and converts it into the digital domain using a 24-bit $\Delta\Sigma$ Analog-to-Digital converter (ADC), which in turn uses a second-order modulator optimized for low-noise applications [42]. The input signal is sampled by the modulator at the rate of $f_{\text{MOD}} = \frac{f_{\text{CLK}}}{2}$. The digital filter then receives the modulator output and decimates the data stream. Trade-offs can be made between resolution and data rate by adjusting the amount of filtering [42]. The filter applied is a Sinc filter: a variable decimation rate, third-order, low-pass filter.

All channels are first passed through an electrostatic discharge (ESD) protection circuit to protect the BIADC from any electrical shorts with any object that comes into contact with the channel pins. Signals are then fed to a voltage divider to reduce the voltage of the incoming signals. The reduced, analog signals are then fed into the BIADC for conversion. XTREMIS utilizes a Texas Instruments ADS1299 chip for the BIADC due to its ability to process EKG, EMG, and EEG signals with minimal adjustments. Once the analog signal has been converted to a digital one by the BIADC, it is ready for actuation by the data processing circuit.

3.2.2 Data Processing Circuit (DPC)

While the DAC is responsible for converting biosignals from analog to digital, the DPC is responsible for the application program interface (API) of XTREMIS as well as processing user commands. More specifically, a 32-bit microprocessor is responsible for saving, transmitting, and annotating the data as well as adjusting the settings of the BIADC as per the user's commands coming in from the WiFi SoC. Communications between the actuation components (BIADC, IMU, and SD card) and the processor are done through Serial Peripheral Interface (SPI) protocol so as to handle the high sampling rates that the BIADC and IMU (which contains both a gyroscope and an accelerometer) are capable of.

On the other hand, the WiFi SoC has its own embedded processor and as such communication with the on-board microprocessor was chosen to be Universal Asynchronous Receiver-Transmitter (UART) protocol due to its asynchrony. Synchronization and timing information is embedded into the data stream and synchronization at each end is achieved with a protocol that incorporates start and stop bits. Further, using UART simplifies the design when using streaming-mode: the processor simply dumps the samples collected onto the UART bus and the WiFi SoC picks them up and transmits them. Utilizing UART instead of SPI in this case is namely a design choice as it is sufficient to support the high data rates coming from the IMU and the BIADC, thus SPI was not necessary. Finally, the WiFi SoC is programmed independently from the on-board microprocessor, making the programming of XTREMIS a two-step process.

3.2.3 Data Marking System (DMS)

As with any data collection system, it is paramount that the data can be properly marked and annotated in real time if there is a need for it. Certain tasks like typing cannot simply be marked manually, and require a low latency data marking system to properly mark the ground truth data. Past works and current products lack DMS circuits, making the data collection and training process for machine learning applications cumbersome. Due to the nature of biosignals, it is impossible to mark data without an external device. Data marking is usually performed with a video recording of the subject performing and then cross-referencing it with the biosignal data. The DMS in XTREMIS allows for the marking of data “on-the-fly” – that is, as it is being recorded in real time. The inspiration of the DMS of XTREMIS came from trying to mark the data coming from a research-grade EMG machine in our previous work [89]. To resolve this, we constructed a DMS (referred to as a trigger mechanism) using an Arduino and a PS/2 keyboard that interfaced with the EMG machine to properly mark the data when a key was pressed and when it was released.

XTREMIS employs a similar mechanism in which 6 pins are dedicated as input pins, the state of which determines whether or not there needs to be a marking on the data (they are always defaulted to 0 when there is no incoming marker). The reason for using 6 pins (which correspond to 6 bits in software) is to allow a high variability in possible marking mechanisms. Specifically, 6 bits is enough to represent the alphabet in lower case ASCII. Hence it is possible to mark the data using letters from the English language. If one of the pins’ states is set to 1, then the data is marked in that instance and the states are cleared until the next change. This makes it simple to attach any trigger mechanism to any or all of the pins on XTREMIS that changes the states of the pins appropriately according to an action taken by a user, thereby simplifying the collection of data.

3.2.4 Daisy Chaining

In some applications, 8 data channels are not enough to capture all the relevant signals. This is common in instrumentation that has a high amount of data output, such as brain wave analysis using EEG. The BIADC module on XTREMIS is a Texas Instruments 8-channel ADS1299 analog-to-digital converter that is capable of connecting to another ADS1299 to double the channel output from 8 to 16 channels. Figure 3.7 shows the daisy chain configuration using two ADS1299 chips. The Daisy box represents either another board that contains an ADS1299 directly interfaced with XTREMIS or simply another XTREMIS board. When in daisy-chain mode, both the daisy and XTREMIS share the clock (SCLK), chip select (CS), and data-in (DIN) pins. The data-out (DOUT) of the daisy is connected to the daisy input (DAISY˙IN) of XTREMIS. Since 16-channel setups are typically utilized for EEG signals, we leave the analysis, experimentation, and verification of the daisy-chained 16-channel XTREMIS to future works due to it being out of the scope of this work.

3.3 Classification

Raw EMG signals come in a somewhat useless form. It hence becomes important to preprocess and analyze them before training a classifier. The raw signal is first

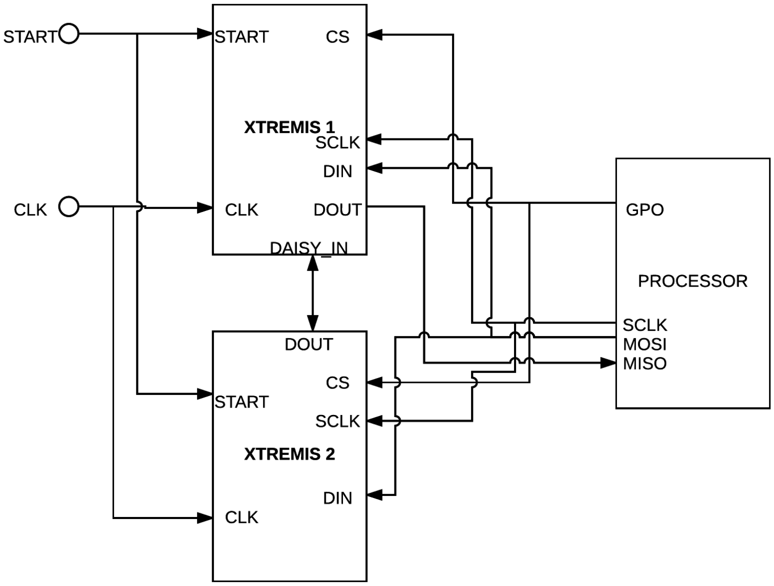


Figure 3.7: Daisy chaining XTREMIS using the ADS1299 [42].

cleaned up, and relevant segments of the clean signal are then extracted. Features are then extracted and used in training or invoking a classification.

3.3.1 Preprocessing and Windowing

The first stage that EMG signals have to go through is data preprocessing. The ISEK Standards of Reporting EMG Data states that the firing rate of Motor Unit Action Potentials (MUAP) is typically between 10Hz and 500 Hz [60], and as such the EMG data is first passed through a bandpass filter within 10Hz and 500Hz. In regular gesture recognition systems the next step is usually segmentation, which involves separating inactive periods from active periods in the signal. Active periods are defined as blocks in the time-series signal where muscle contractions are happening, while inactive periods are blocks where the muscle is relaxed. However in ballistic gestures resting periods are very brief or sometimes non-existent. Instead, they become *transition* periods, which are times where the finger is moving from a key release to the next key press. Since transition periods vary in intensity across people and typing speeds, we use overlapping windows to label the onset of key press, period where the key is pressed, and key release.

In regular gesture recognition literature, there are several approaches to segment EMG signals [12, 48, 106, 113], the most common of which are:

1. Segmentation by detecting peaks of Motor Unit Action Potentials (MUAPs).
2. Segmentation using energy/peak detection.
3. Segmentation using Discrete Wavelet Transforms.

Although their performance is good when applied to regular gesture recognition, a caveat of the above methodologies is that they require calculation of parameters such as thresholds, Maximum Voluntary Contraction (MVC), and appropriate window sizing, usually found using trial-and-error in experimentation. Further, using thresholding may be counter intuitive as the instability of signals may create inconsistencies between the shapes of active and transition periods. Although the peaks in the waveform are evident, the fluctuations of rest periods makes it difficult to detect them reliably. These fluctuations are referred to as "movement epenthesis", which occur when performing rapid movements. This is common when dealing with ballistic gestures due to the spontaneity of the gestures and their similarities to a muscle twitch, making them difficult to remove completely using only filters. Finally, thresholds are likely to differ when applied to different people as everyone has different physiologies due to factors like the amount of body fat they have and their muscular structure. Hence, the period between a key release and the next key press cannot be treated as a resting period. Instead, using overlapping windows to label onsets, offsets, and transitions between keys will help to build more robust classifiers. Features are then extracted from each window and the corresponding label that is within it (i.e. a key press, release, or in between a press and a release) is used as the ground truth.

3.3.2 Windowing

When extracting data for training, a 50ms sliding window approach with a 10ms overlap was adopted in which if a key press marker is found then that window and all subsequent windows until the key release marker are labeled with the finger mapped to that key. The benefits in using the sliding window in training the HMM is two-fold: First, a sliding window-style of labeling gives information as to how long finger presses and transitions are, making it more resilient to typing speeds. Second, sliding windows allow for training the HMM to detect onsets and offsets of key presses, making it more resilient to movement epenthesis – a common phenomenon when dealing with ballistic gestures that occurs due to the spontaneity of the gestures and their similarities to a muscle twitch, making them difficult to remove completely using a filter [89].

3.3.3 Feature Extraction

Once windows of the signal have been identified, features are extracted from each window. The features selected in this work were chosen from past gesture classification work [12, 39, 52, 76] on fine-grained gesture recognition such as playing the piano [52] and sign language [12]. Alternatively, a random-forest classifier can be used to find the best performing features. Two types of features were extracted: time domain (TD) features and frequency domain (FD) features.

Time Domain Features

Some of Hudgin's feature set was utilized to obtain the time domain features due to the simplicity of their computations and their ability to describe the signal at the time domain well [39]. In this work, four of these features were implemented in addition to the Root Mean Square. Samples are first divided into overlapping windows of length N each. Let I be the total number of windows.

1. *Mean Absolute Value (MAV)* — Estimate of the MAV of the signal in window i which is N samples in length, given by:

   $$\mathrm{MAV}_i = \frac{1}{N}\sum_{k=1}^{N}|x_k^i| \quad \text{where } i = 1, ..., I \tag{3.1}$$

   where x_k^i is the k^{th} sample in window i and I is the total number of windows over the entire signal.

2. *Difference MAV* — Represents the difference in MAV between the window i and the subsequent window $i+1$, given by:

   $$\Delta\mathrm{MAV}_i = \mathrm{MAV}_{i+1} - \mathrm{MAV}_i. \tag{3.2}$$

3. *Slope Sign Changes (SSC)* — Number of times the slope of the waveform changes signs (from positive to negative, or vice versa). A suitable threshold

must be chosen to reduce noise induced changes. Given three consecutive samples in window i: x^i_{k-1}, x^i_k, and x^i_{k+1}, the slope sign change count is incremented if:

$$\begin{aligned} x^i_k > \max(x^i_{k-1}, x^i_{k+1}) \quad \text{or} \quad x^i_k < \min(x^i_{k-1}, x^i_{k+1}) \\ \text{and} \quad \max(|x^i_{k+1} - x^i_k|, |x^i_k - x^i_{k-1}) > \Omega, \end{aligned} \tag{3.3}$$

where Ω is a threshold value that is determined as 0.02mV for a noise value of 21.34μV peak-to-peak [42].

4. *Waveform Length (WL)* — The cumulative length of the waveform over the window i. This is the cumulative length of the waveform over the time window, defined as:

   $$\text{WL}_i = \sum_{k=2}^{N} \left| x^i_k - x^i_{k-1} \right|, \tag{3.4}$$

   where $x^i_k - x^i_{k-1}$ is the difference in consecutive sample voltages [39].

5. *Root Mean Square (RMS)* — Provides a measure on the power of the signal in window i which is N samples in length.

   $$\text{RMS}_i = \sqrt{\frac{1}{N} \sum_{k=1}^{N} (x^i_k)^2}. \tag{3.5}$$

Frequency Domain Features

The frequency domain features used were Hjorth's parameters, the mean, and median frequencies extracted from the power spectrum of the signal. Hjorth's parameters are normalized slope descriptors typically used in EEG processing for data reductions or automatic sleep stage scoring. In this work we use them for the detection of the transient EMG signal which occurs during muscle contractions, or when a muscle is switching from a relaxed to a contracted state and vice versa. We first define the spectral moments over a discrete fourier transform. The zero-order moment m_{0_i} over window i is proportional to the mean energy in that window, and is defined as:

$$m_{0_i} = \frac{1}{F} \sum_{f=f_1}^{f_2} P_f^i, \tag{3.6}$$

where f_1 to f_2 are a range of frequencies of length $F = f_2 - f_1$, and P_x^i is the discrete power spectrum of the signal in window i. Since the EMG frequency range is between 0Hz and 500Hz [60], the range $f_1 = 0$ and $f_2 = 500$ were used. The first-order moment is defined as:

$$m_{1_i} = \frac{1}{F \cdot m_{0_i}} \sum_{f=f_1}^{f_2} P_f^i\left(\frac{f}{N_i \Delta t}\right), \tag{3.7}$$

where N_i is the length of the power spectrum, and Δt is the sampling interval. To get higher order shape information one needs to define the central moments in the discrete domain as follows:

$$m_{n_i} = \frac{1}{F \cdot m_{0_i}} \sum_{f=f_1}^{f_2} P_f^i\left(\frac{f}{N_i \Delta t} - \bar{f}\right)^n, \tag{3.8}$$

where $\bar{f}$ is the normalized value found from the discrete first order moment. Now that the spectral moments have been defined, we can proceed to defining the Hjorth parameters:

1. *Activity* — Represents signal power as the variance of the amplitude of the signal. In the frequency domain, it can be conceived as the envelope of the power spectrum [63] in window i. Activity is defined as:

   $$\text{Activity}_i = m_{0_i} = \sigma_{0_i}^2, \tag{3.9}$$

   where $\sigma_{0_i}^2$ is the variance of window i.

2. *Mobility* — measures the ratio between the standard deviation of the slope and the standard deviation of the amplitude per time unit. In other words, it represents dominant frequency. Mobility for window i is defined as:

   $$\text{Mobility}_i = \sqrt{m_{2_i}/m_{0_i}} = \sigma_{1_i}/\sigma_{0_i}, \tag{3.10}$$

   where the second order moment m_{2_i} is a measure of the width of the spectrum about the mean in window i, and σ_{1_i} is the standard deviation of the first derivative of the signal in window i.

3. *Complexity* — Represents change in frequency. It compares the signal's similarity to a pure sine wave [63], and is defined as:

   $$\text{Complexity}_i = \sqrt{(m_{4_i}/m_{2_i}) - (m_{2_i}/m_{0_i})} = \frac{\sigma_{2_i}/\sigma_{1_i}}{\sigma_{1_i}/\sigma_{0_i}}, \tag{3.11}$$

where σ_{2_i} is the standard deviation of the second derivative of window i.

Finally, we obtain the mean and median frequencies from window i as follows:

1. *Mean Frequency* — an average frequency which is calculated as the sum of the product of the EMG power spectrum and frequency divided by the total sum of the power spectrum [68, 73].

$$f_{\mathrm{mean}_i} = \frac{\sum_{j=1}^{M} f_j \cdot P_j^i}{\sum_{j=1}^{M} P_j^i}, \tag{3.12}$$

   where M is the number of frequency bins, f_j is the frequency value of the EMG power spectrum at frequency bin j, and P_j is the EMG power spectrum at frequency bin j.

2. *Median frequency* — a frequency at which the EMG power spectrum is divided into two regions with equal integrated power[68, 73]. The halved power spectrum is then traversed and the median frequency is obtained.

$$\sum_{j=\mathrm{f}\dot{}\mathrm{median}}^{M} P_j^i = \frac{1}{2} \cdot \sum_{j=1}^{M} P_j^i, \tag{3.13}$$

   where MDF is the median of the number of frequency bins M.

3.3.4 Hidden Markov Model

To take advantage of the patterns produced by fingers while typing, the HMM is employed to predict the fingers that have been pressed given time t. A left-to-right Gaussian mixture HMM is built with 6 states: 5 fingers and a transition state, which represents the hand movement as it is between releasing a key and pressing the next key. Figure 3.8 shows the state transition diagram between the finger and transition states.

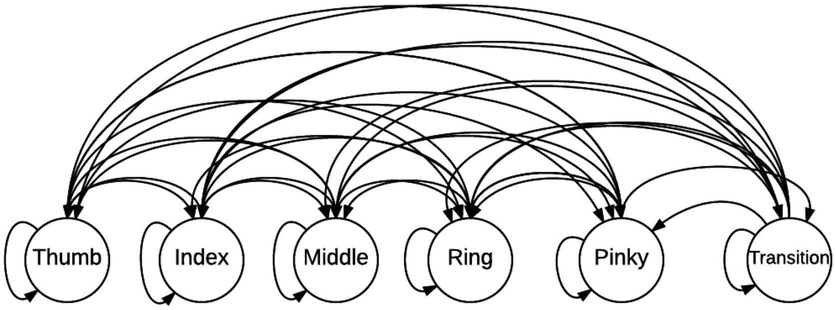


Figure 3.8: State transition diagram of the GM-HMM.

A HMM $\lambda = (A, B, \Pi)$ consists of the following:

1. A set of H hidden states, $S = S_1, S_2, ..., S_N$.
2. A state transition probability distribution $A = a_{ij}$ where $i, j \in S$. More formally,
$$a_{ij} = P(S_t = S_j | S_{t-1} = S_i), 1 \leq i, j \leq H \tag{3.14}$$
3. An observation probability distribution $B = b_i(o_t)$. The probabilistic function for each state s_i is:
$$b_i(o_t) = P(o_t | S_t = S_i) \tag{3.15}$$
where o_t is the observation at time t.
4. An initial probability distribution $\Pi = \pi, i \in S$ where π_i is defined as:
$$\pi_i = P(S_1 = S_i) \tag{3.16}$$

The transition and observation probability distributions A and B are constructed from training data. The state relationships can be thought of as a fully connected graph, in which a state has a non-zero probability of transitioning to any other state as shown in Figure 3.8. However, due to the abundance of transition states (users are more often in transition than using any other finger) every finger state has a higher probability of transitioning to a transition state than a finger state.

In a GM-HMM, the function $b_i(o_t)$ takes the form of a Gaussian mixture of continuous probability density functions (PDF):

$$b_i(o_t) = \sum_{k=1}^{M} w_{ik} b_{ik}(o_t), i = 1, ..., N \tag{3.17}$$

where M is the number of mixtures and w is the weight of each mixture. The mixture weights have the following constraints:

$$\sum_{k=1}^{M} w_{ik} = 1; w_{ik} \geq 0, i = 1, ..., N, k = 1, ..., M \tag{3.18}$$

Each $b_{ik}(o_t)$ is a d dimensional Gaussian density with mean vector μ_{ik} and covariance matrix Σ_{ik}. For each state, a multivariate Gaussian density in the form:

$$b_{ik}(o_t) = g(o_t, \mu_{ik}, \Sigma_{ik}) = \frac{1}{\sqrt{((2\pi)^d |\Sigma_{ik}|}} \cdot exp\Big(-\frac{1}{2}(o_t - \mu_{ik})^T \Sigma_{ik}^{-1}(o_t - \mu_{ik})\Big) \tag{3.19}$$

is used where o_t is the observation. For each state s_i, the mean and covariance (μ_i, Σ_i) are used to construct the emissions matrix. Hence, it is important to choose an appropriate value for d for each state's Gaussian mixture density function, which in turn defines its mean and covariance pairs. To do so, a GMM with different parameters is fit to the observations of each state. The model with the smallest Bayesian Information Criterion (BIC) is selected as the most representative model of the data. Indeed, minimizing the BIC corresponds to maximizing the posterior model probability for a large number of observations and is an effective method of selecting a model [108]. The BIC can be defined as:

$$BIC = -2 \cdot \ln(\hat{\theta}) + p \cdot \ln(n), \tag{3.20}$$

where $\hat{\theta} = p(\hat{o}|\delta, M)$ is the maximized value of the likelihood function of the model M, p is the number of free parameters to be estimated, and n is the number of observations [108].

3.4 Experimental Setup

To verify the functionality and fidelity of XTREMIS, its signal is compared to the gold standard in ECG/EMG/EEG data collection: the BioSemi ActiveTwo [10]. Specifically, XTREMIS is compared to BioSemi in two aspects:

1. **Signal quality**: We evaluate the signal of XTREMIS as well as compare it to BioSemi's signal in a salt water experiment to verify their similarity.
2. **GM-HMM classification performance**: We compare XTREMIS to BioSemi in ballistic gesture recognition in a similar experiment to our previous work [89]. In this work, however, the experiment encompasses only anatomically-aware electrode placement using XTREMIS and BioSemi ActiveTwo at the same

1024Hz sampling rate, whereas the previous work explored a ring-of-electrodes configuration and different sampling rates.

We begin by presenting the signal-level analysis and a signal-to-noise ratio comparison between XTREMIS and BioSemi ActiveTwo, then proceed to show that the highly correlated signal leads to a good ballistic gesture classification accuracy by performing typing experiments on 8 participants: 4 females and 4 males.

3.4.1 Classification Comparison

Finger movement prediction is analyzed during a typing task under 3 different typing speeds: slow, regular, and fast. A 500-word paragraph is presented to the subject. The experimental procedure is as follows:

1. The subject is asked to perform a test run of the paragraph.
2. The subject types the paragraph at regular speed (10-25 words-per-minute).
3. The subject then types the paragraph at a fast typing speed (30-50 words-per-minute).
4. The subject types the paragraph as fast as they can (50-75 words-per-minute).

Additionally, finger movement is analyzed during the typing task under 3 factors:

1. Words-per-minute (WPM).
2. Electrode placement configurations.
3. Sample rates.

The BioSemi and XTREMIS are fitted onto subjects in an alternating fashion such that half the subjects had XTREMIS fit onto them first, and the other half had BioSemi fit first. This acts as a control in case subjects' typing habits change as they type a more familiar paragraph (e.g., they're devoting fewer cognitive resources to reading the paragraph, and hence are more able to exert control over which fingers they're using). It is important to note that both systems used the same type of electrodes – silver/silver chloride. After each time the subject types the paragraph, they have the option of taking a rest to recuperate and ensure they are not too tired to continue. We will first discuss the process of marking ground truth labels for each system.

Automated Collection of Ground Truth Labels

As discussed in Section 3.2.3, XTREMIS' DMS works similarly to BioSemi's trigger interface. To properly annotate data with key press and key release times, an external trigger system was constructed to translate key events to data markers. Therefore, two systems were built that functioned almost identically: a trigger system for BioSemi, and one for XTREMIS. Each system consisted of a PS/2 keyboard interfaced with an Arduino Uno, which in turn was interfaced with the BioSemi trigger input connector or the DMS on XTREMIS. Figure 3.9 shows the setup for BioSemi. In a sense, the Arduino functions as a Serial-to-Parallel converter: it receives input from the PS/2 keyboard, converts it to a binary number (8-bits for BioSemi, 6-bits for XTREMIS), and then inputs this number to BioSemi's trigger system or XTREMIS' DMS.

Further, electrodes were placed on the individual's right forearm in an anatomical configuration identified in previous works to be the best placement of electrodes to

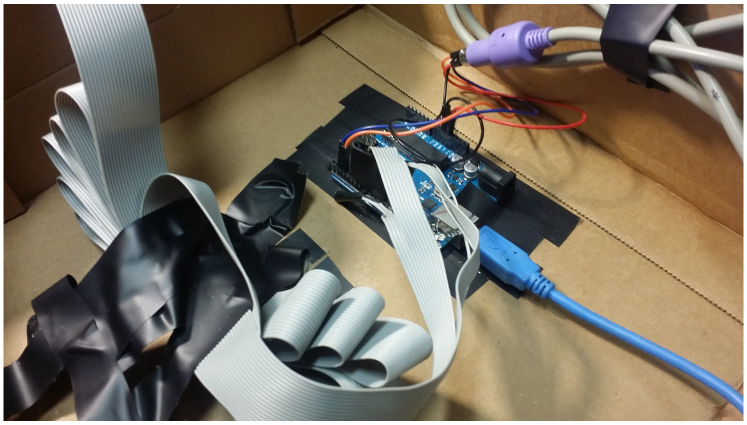


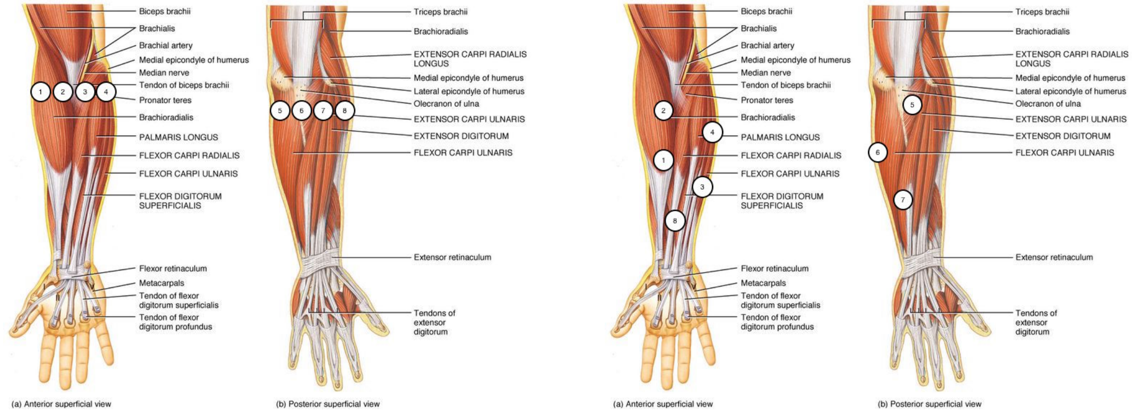
Figure 3.9: Arduino-controlled trigger system. The colored wires belong to the keyboard whereas the gray ribbon wires plug into the BioSemi ActiveTwo's trigger input (triggers are labels inserted into the system from an external device)

capture muscular contractions clearly [44, 89]. Figure 3.10(b) highlights where the 8 electrodes were placed in both the BioSemi and XTREMIS setups.

3.5 Results and Discussion

3.5.1 Signal Quality

To verify the SNR of XTREMIS over different frequencies, a signal generator was used to input sine wave signals of varying frequencies at different sampling rates. The SNR was calculated by first normalizing the and extracting 5 segments – equally spaced – from the entire data stream. The measured signal may not always be strictly periodic (due to various potential factors surrounding the experiment), making it more similar to a real biomedical signal. If this irregularity is left unchecked, it will create discontinuities that appear as high frequency components in the fast Fourier transform (FFT). Hence for each segment, a Hann window [33] is extracted to reduce the amplitude of these potential discontinuities. The power of the signal is then



(a) (b)

Figure 3.10: (a) Myo's rigid placement of electrodes in a Ring formation. (b) Anatomically aware placement of electrodes used by XTREMIS and BioSemi ActiveTwo.

obtained from the FFT using Welch's method [104] since the frequency range of the fundamental frequency of the input signals is known. The SNR is then obtained from the PSD as follows:

$$SNR = 10 \cdot log_{10}\left(\frac{\sum\limits_{i=f_1}^{f_2} P(i)}{\sum_{f\in F} P(f) - \sum\limits_{i=f_1}^{f_2} P(i)}\right), \tag{3.21}$$

where $P(i)$ is the normalized power of the signal between frequencies f_1 and f_2 and $P(f)$ is the normalized power of the signal at all frequencies. Frequencies f_1 and f_2 were the surrounding frequencies on either side of Figure 3.11 shows the SNR of the XTREMIS signal at different sine wave frequencies into one channel of XTREMIS collecting at 1024Hz.

It should be noted that the ADS1299 chip's sample rate becomes less stable at higher frequencies. This is likely due to the design of the ADS1299 chip combined with the small form factor of XTREMIS, as the wires on board may generate high

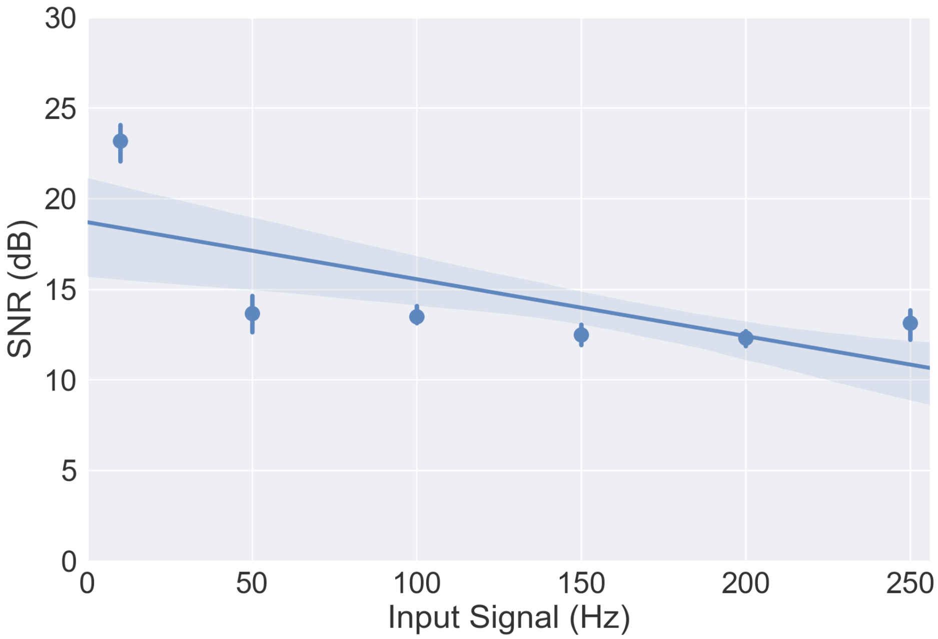


Figure 3.11: SNR of XTREMIS signal at different input signal frequencies.

frequency noise that is picked up by the chip. It was found that approximately 1024Hz is the maximum sampling rate at which the SNR is high enough to obtain a reliable signal at all biosignal wave frequencies. Since the bandwidth of EMG, EEG, and ECG signals is 10-500Hz [60], 1-50Hz [83],and 0.05-100Hz [103] respectively, the signal quality of XTREMIS is still sufficient at high sampling rates to capture more than the Nyquist frequency of the ECG, EEG, and EMG signals.

When sampling at rates higher than 500 Hz and high gains (specifically, a gain of ×8), there is a small shift in the detected frequency of a given signal due to a higher noise. Figure 3.12 shows the percentage in which the signal frequency shifts at different input signal frequencies while sampling at 1024Hz. This measure can also be considered as a percent error on the detected signal frequency with different input signals. Since we already know the PSD of the incoming signal, it is possible to calculate the signal frequency shift as follows:

$$shift = 100 \times \frac{f_{max} - f_{input}}{f_{input}}, \tag{3.22}$$

where f_{max} is the frequency detected by XTREMIS with the highest amplitude and f_{input} is the frequency of the input signal. The similarity in shift percentage at different frequencies indicates that there is a consistency to the frequency shift. This consistency is due to the fact that the ADS1299 chip always samples at its highest frequency (16kHz) regardless of sample rate. Indeed, changing the sampling rate on the ADS1299 in fact changes the decimation ratio, and hence why at higher sampling rates there is a higher amount of noise due to the reduction in averaging samples. This also corresponds with the ADS1299 data sheet specification regarding the increase of noise if the data rate and the gain setting are set too high. Data rates are to be minimized for each application to reduce noise as much as possible [42].

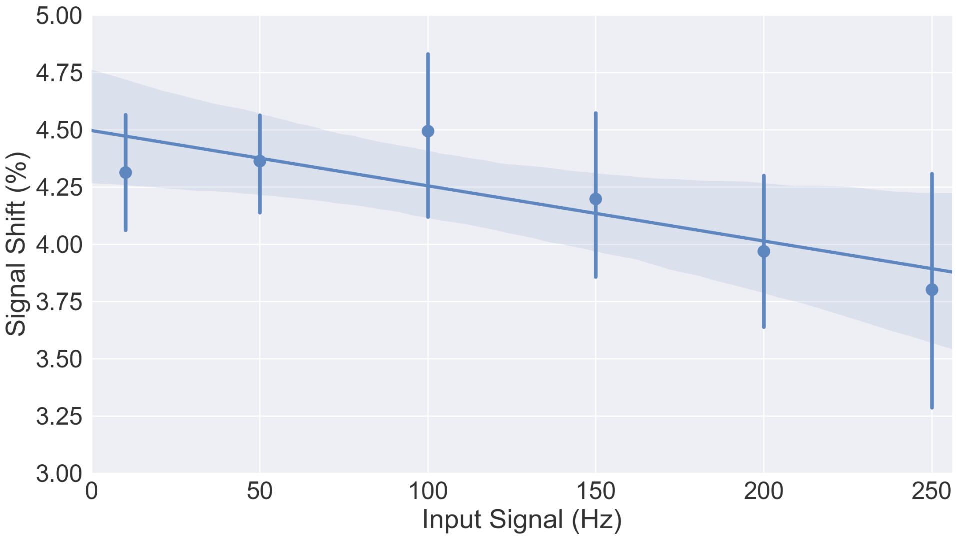


Figure 3.12: Shift of XTREMIS signal at different frequencies.

3.5.2 Signal Comparison

Since BioSemi ActiveTwo is the gold standard of ECG/EMG/EEG machinery, it is important to ensure that the signal from XTREMIS is similar to the signal from BioSemi. To analyze this, electrodes from a signal generator are placed in two buckets of saline solution (a mixture of sodium chloride with water). The reference electrodes are placed into one bucket(the reference bucket) and the signal electrodes are placed into the other (signal bucket). Due to the dry surface electrodes and the closed form factor of Myo, it was not possible to include it on this test.

The reference electrodes of XTREMIS and BioSemi ActiveTwo are also placed into the reference bucket, while their channel electrodes are placed into the signal bucket. The signal generator then generates a sine wave signal at a predetermined frequency. XTREMIS and BioSemi are configured to collect data from the electrodes submerged in the signal bucket and their signals are then compared. This experimental setup is common in testing EMG equipment due to the similarity in electrical conductivity between the human body and saline. Figure 3.13 highlights the setup in separating the reference electrodes and the channel electrodes in two separate buckets.

Figure 3.14 shows the SNR calculated for XTREMIS and for BioSemi ActiveTwo in this experiment. The SNR of BioSemi ActiveTwo is – as expected – higher than XTREMIS as it is a research-grade ECG/EEG/EMG machine with more sophisticated and proprietary biomedical technology. However both systems follow a similar trend in their SNR progression at different frequencies, indicating that XTREMIS is a valid alternative due to its portability, ease-of-use, and robust signal.

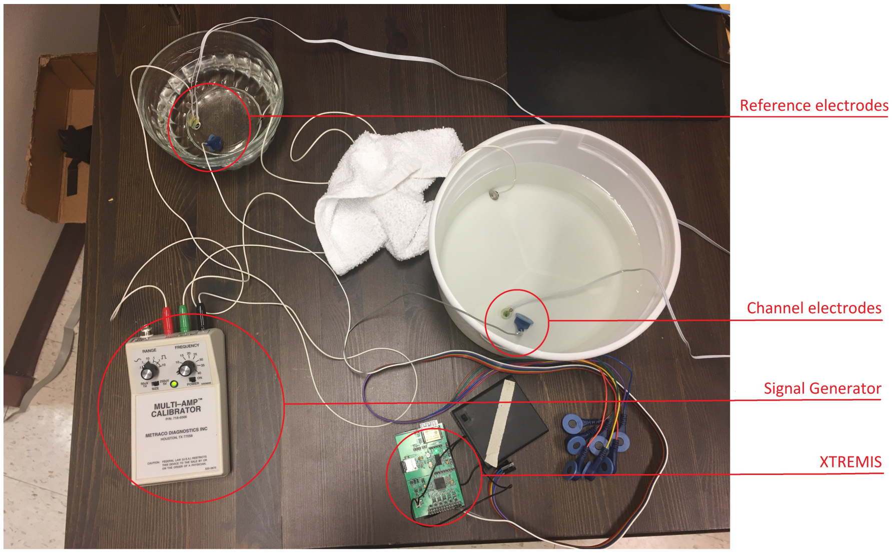


Figure 3.13: The setup for the salt water signal comparison test. Reference electrodes from the signal generator, XTREMIS, and BioSemi are placed in the bucket to the top left, while the channel electrodes are placed in the bucket to the right.

3.5.3 Classification Results

We compare the classifier performances averaged over all subjects across different speeds using ground truth labeling described in Section 3.4.1 and following the experimental procedure discussed in Section 3.4.1. Figure 3.15(a) shows the average accuracy of all subjects across 3 different speeds while collecting data from BioSemi, XTREMIS, and the Myo armband. The similarity in the accuracies suggests that the signal of BioSemi and XTREMIS is similar enough that it is possible to have comparable performance in classification. Further, this result is significantly higher than the highest accuracy achieved by Myo: approximately 82% at the slowest typing speed.

A strong classification model should function across speed as well as users. Figure 3.15 (b) highlights the performance of the GM-HMM user-independent (UI) model

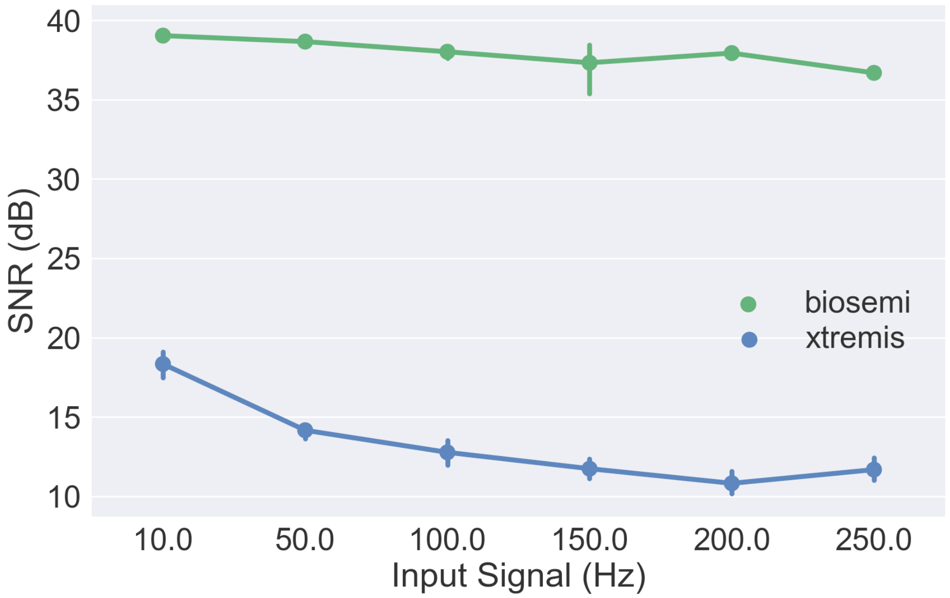
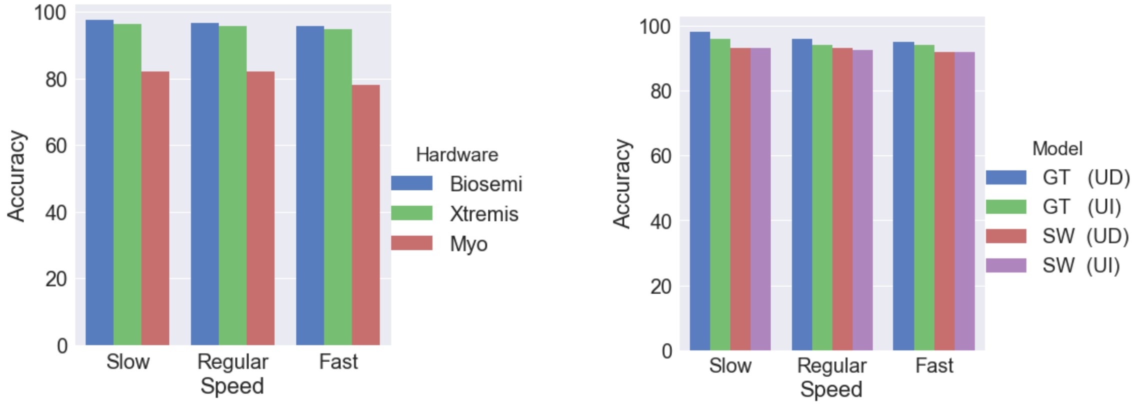


Figure 3.14: Comparison between the SNR of XTREMIS and BioSemi ActiveTwo.

v.s. the user dependent (UD) model on data collected using XTREMIS. The accuracy of UD model reported is the average accuracy from all user trials. The UI GM-HMM generally performs almost as well as its UD counter-part, indicating that the GM-HMM can be trained to function across multiple users using XTREMIS.

Table 3.1 highlights the average precision and recall achieved across all users using ground truth data on all systems. The consistent high precision and recall scores across all speeds as well as their similarity between XTREMIS and BioSemi indicate that not only does the GM-HMM function well across speeds, but XTREMIS' performance and data quality collected is similar to BioSemi. On the other hand, the Myo armband's precision and recall suffer due to its lower sample rate and rigid form factor.

Tables 3.2 and 3.3 are confusion matrices of the GM-HMM trained on BioSemi and XTREMIS, respectively. The GM-HMM suffers most on both devices when



(a) (b)

Figure 3.15: (a) GM-HMM accuracy using ground truth data using different hardware. Results averaged over all subjects. (b) XTREMIS Accuracies across User-Dependent and User-Independent GMHMM models

Table 3.1: Average Precision and Recall when typing a paragraph using XTREMIS v.s. BioSemi ActiveTwo v.s. Myo armband

		XTREMIS		**BioSemi**		**Myo Armband**	
		Precision	Recall	Precision	Recall	Precision	Recall
Speed	Slow	92.4%	88.2%	95.2%	90.2%	78.16%	72.31
	Regular	93.1%	89.1%	94.8%	91.3%	77.02	73.47
	Fast	91.8%	89.0%	93.1%	89.7%	74.45	64.33

classifying between middle and ring fingers. This may be due to the middle and ring fingers sharing several muscles and as a result having a very similar waveform. The lower classification on XTREMIS suggests that the signal quality is a contributing factor to classification, even though the GM-HMM also takes advantage of movement patterns during the training phase.

Table 3.2: Confusion matrix of GM-HMM using XTREMIS

		Ground Truth					
		Thumb	Index	Middle	Ring	Pinky	Transition
Prediction	Thumb	96%	0%	0%	0%	0%	4%
	Index	0%	91%	7%	0%	0%	2%
	Middle	0%	0%	92%	5%	0%	3%
	Ring	0%	0%	6%	94%	0%	0%
	Pinky	0%	0%	0%	5%	92%	3%
	Transition	0%	0%	0%	0%	0%	100%

Table 3.3: Confusion matrix of GM-HMM using BioSemi ActiveTwo

		Ground Truth					
		Thumb	Index	Middle	Ring	Pinky	Transition
Prediction	Thumb	97%	0%	0%	0%	0%	3%
	Index	0%	94%	4%	0%	0%	2%
	Middle	0%	0%	92%	5%	0%	3%
	Ring	0%	0%	4%	95%	0%	1%
	Pinky	0%	0%	0%	5%	93%	2%
	Transition	0%	0%	0%	0%	0%	100%

3.5.4 Configuration Effects

Prior to constructing XTREMIS, we compared the classifier performances averaged over all subjects across different speeds using the ring and anatomical electrode placements using subject dependent training on Myo and BioSemi ActiveTwo. Table 3.4 shows the classification accuracies obtained when typing a paragraph using the Myo with ground truth data.

Table 3.4: Classifier performance when typing a paragraph while wearing the Myo armband.

		Paragraph Typing			
		HMM	SVM	kNN	DT
WPM	Slow	82%	65.3%	63%	63.2%
	Regular	82%	55.3%	53.1%	53%
	Fast	80%	52.3%	49.8%	48.1%
	Fastest	78%	43%	42.2%	41%

Table 3.5: Classifier performance when typing a paragraph using the Biosemi ActiveTwo sEMG, with electrodes arranged in a ring configuration like the Myo armband.

		Paragraph Typing			
		HMM	SVM	kNN	DT
WPM	Slow	94%	72%	72%	70%
	Regular	93.2%	70%	69.2%	67%
	Fast	93%	68%	66.5%	64.2%
	Fastest	92.6%	66.3%	66%	64.1%

Table 3.6: Classifier performance when typing a paragraph using the Biosemi ActiveTwo sEMG, with electrodes placed on specific muscles as shown in Figure 3.10(b)

		Paragraph Typing			
		HMM	SVM	kNN	DT
WPM	Slow	98.4%	88.2%	87.4%	85.2%
	Regular	97.1%	85.1%	84.2%	84%
	Fast	96.8%	76%	74.5%	74%
	Fastest	96.4%	71%	69.4%	68.2%

Tables 3.5 and 3.6 show the average performance over all subjects of each classifier for the Biosemi ActiveTwo system in ring and anatomical-based configurations, respectively. The GM-HMM's consistency of performance across speeds and degradation of performance across placements show that electrode placement indeed plays a role in classification accuracy. Moreover, using a GM-HMM with an anatomical placement of electrodes yields the best results. The confusion matrix in Table 3.2 shows that while the HMM with anatomical placement of electrodes performs well, it still suffers most when classifying between middle and ring fingers. This may be due to the middle and ring fingers sharing several muscles and as a result having a very similar waveform. Other classifiers' performance drops significantly as speed increases, likely due to their inability to take advantage of transition probabilities.

Additionally, the high performance of the Biosemi ActiveTwo compared to the

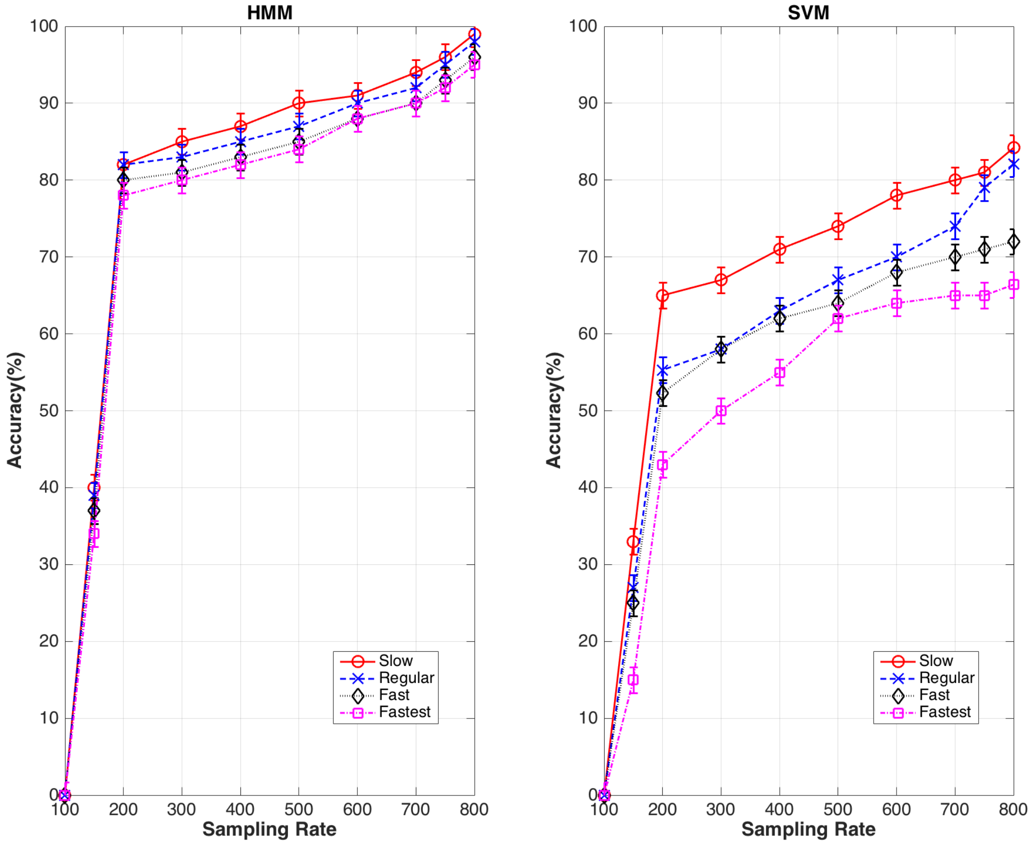


Figure 3.16: Classifier accuracy V.S. sampling rate using Biosemi ActiveTwo with anatomical configuration for HMM and SVM.

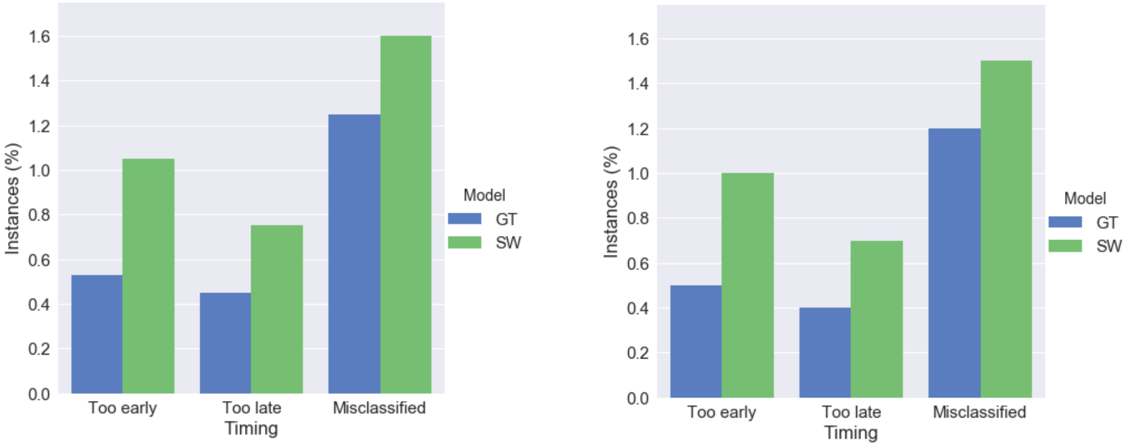
Myo indicates that sampling rate also plays a role. To verify this, the data obtained from the anatomical placement was downsampled down to approximately 100Hz and the classifier accuracy was obtained at each sampling rate. Figure 3.16 shows the changes in HMM and SVM accuracy as sampling rate increases. For both methodologies, a sliding window of 50ms with 10ms overlap was used. Window size and sampling rate are correlated in that if one is sufficiently large, the other must be sufficiently small. In other words, the window size must be larger in order to include more samples for analysis. Similarly, if the sampling rate is high, the window size should be smaller. Due to timing constraints, we have not investigated the impact of different window sizes on accuracy, and leave it to future works. Additionally, at 200Hz the performance becomes comparable to the accuracies achieved by Myo in Table 3.4. Therefore, this validates the ISEK standard [60] for EMG signal bandwidth as well as confirms that the 200Hz claimed by Myo is not enough for research-level use [88].

Onset Detection

Finally, Figures 3.17(a) and (b) show a comparison between the timings of classifications for two GM-HMMs for each device: one using ground truth (GT) as training and another using the sliding window (SW) approach. A “too early” classification indicates that the GM-HMM classified a window as a finger when it was still a transition, while a “too late” classification indicates that the GM-HMM classified a window as a transition when it has become a finger already. Finally, a “misclassification” is when the GM-HMM misses a finger movement entirely or classifies a finger as another finger. Therefore, finger detection accuracy is not only defined as correct v.s. incorrect classifications, it is also defined by the *timing* of classifications. Figures 3.17(a) and (b) indicates that the GM-HMM is more inclined to make a classification too early rather than too late. This can be due to the delay between a muscle contraction and a key press, as humans must first contract the muscles before a movement is made. Additionally, this result is consistent with the timing analysis performed in our previous work [89].

3.5.5 Power Consumption Analysis

Power consumption analysis is an important cornerstone of any embedded system solution. Due to the Wi-Fi SoC and the ADS1299 chip’s power consumption, the



(a) XTREMIS (b) BioSemi ActiveTwo

Figure 3.17: Timing of GM-HMM Recognition on BioSemi ActiveTwo and XTREMIS

total power consumption of XTREMIS cannot be strictly classified as “low-power”. However, XTREMIS requires only 3 AA batteries connected in series to perform all of its functions. Table 3.7 shows the power consumption and estimated battery life for each operating mode with ADS1299 collecting at 1024 Hz powered by 4 Duracell Procell AA batteries with a capacity of 2148 mAh. These estimates are based on back-of-envelope calculations and hence may not be strictly accurate, however they provide a reasonable rough idea of the expected operation time. The most power-hungry mode is streaming over Wi-Fi at 1024 Hz, which lasts approximately 18 hours – which is still long enough for longitudinal studies like sleep studies. Myo, on the other hand, claims a 24 hour battery life while streaming [41]. However, Myo has an unfair advantage in this case as it is using BLE technology to stream at approximately 250 Hz. Since the ADS1299 collects data at a rate of 16kHz and then decimates it according to the user’s chosen data rate, it makes no difference in power consumption to reducing the sampling rate to save power on XTREMIS.

Table 3.7: Power consumption in different modes of operation on XTREMIS

Operating Mode	**Average Current (mA)**	**Estimated Battery Life**
Wi-Fi ON (No data collection)	115.60 mA	18 hours 38 minutes
Wi-Fi OFF (No data collection)	87.91 mA	24 hours 43 minutes
Wi-Fi ON (Data collection and streaming)	120.43 mA	17 hours 45 minutes
Wi-Fi OFF (Data collection)	113.75 mA	18 hours 53 minutes

3.6 Summary

We presented XTREMIS: a low-cost and portable EMG platform with a similar form factor to off-the-shelf wearable sensors that is comparable to research-grade EMG machines in sampling rates, electrode placement fluidity, and signal-level processing. Indeed, experiments on 8 subjects have shown that produces a signal that is similar to that of research-grade EMG machines, leading to a better classification accuracy than wearable off-the-shelf technologies like the Myo armband. It is also capable of ECG and EEG signal data collection.

Chapter 4

SiCILIA: Inferring Clothing Insulation Levels using Mechanisms of Heat Transfer

Thermal comfort can be defined as the state of mind that expresses satisfaction with the thermal environment [31]. In layman's terms, it means that a person is not feeling too warm or too cold. Thermal comfort is important for health and well-being as well as productivity [107]. An imbalance in thermal comfort will cause stress and discomfort among occupants (that is, humans occupying a space, whether its a room, building, or vehicle) [23]. When occupants are too warm, they may feel fatigued, and when they are too cold, they may feel restless or distracted. These symptoms in turn affect occupant productivity and concentration. Further, in addition to environmental factors such as ambient temperature, humidity, and air speed, personal factors such as clothing insulation and metabolic rate contribute greatly to thermal comfort. Therefore, it is important for Heating, Ventilation, and Air Conditioning (HVAC) engineers to design systems in which occupants will feel as thermally comfortable as possible by taking into account both environmental *and* personal factors.

As it stands, building operators currently use a pre-set indoor temperature for different outdoor temperatures. Temperatures of 22^oC and 24^oC were recommended for summer and winter in ASHRAE Standard 55 [1]. In practice, however, as noted by Gao and Yuan *et al.* [24, 25, 111], building operators tend to deviate from these recommendations to minimize complaints by, for instance, setting the temperature much lower in the summer. However, the current practice to "err on the safe side" is geared for the general population and fails to account for individual needs. Consequently, two problems may arise. First, the resulting set point may lead to discomfort, and in extreme cases the Sick-Building-Syndrome [77]. As championed by De Dear [13], it has been recognized recently that the acceptability of thermal environment is influenced by the occupants' behaviour adaptation, physiological adaptation and psychological adaptation. An example of behaviour adaptation is the addition or removal of clothing. Second, the one-size-fit-all set point would lead to excessive energy consumption due to the typically large difference between outdoor and indoor temperatures. In fact, it has been estimated that approximately 30% to 50% of commercial and residential districts' energy consumption can be attributed to Heating, Ventilation and Cooling (HVAC) systems [11, 25, 72]. Further, about 26 billion liters of fuel (equivalent to about 9.5% of the United States' imported crude oil) are used to annually cool vehicle passenger compartments in the United States [81, 82]. This usage can be reduced significantly with advanced HVAC systems that are aware of the surrounding

---

[1] The recommendation assumes a relative humidity of 50%, a mean relative velocity lower than 0.15m/s, a mean radiant temperature equal to the air temperature, and a metabolic rate of 1.2met. [31, 18, 59].

environmental as well as personal variables of the occupants. Thus, it is also imperative to reduce the energy consumption of HVAC systems while satisfying the personal thermal needs of occupants for better sustainability.

Personal thermal needs are not uniform across all occupants, however. That is, the metabolic rate and, more commonly, the clothing insulation level (i.e. what clothes they are wearing), will not be the same for all occupants. As a result, some occupants may feel warmer or colder than others in the same environment. This can be due to the different levels of clothing insulation worn or due to physiological and psychological preferences. We focus on clothing insulation specifically, as it is one of the most influential factors in determining thermal sensation due to its variety and versatility. Since humans dress according to their preferred sense of thermal comfort (i.e. people who are cold will dress in more layers, whereas people who are warm will dress in fewer layers), it is reasonable to assume that detecting clothing insulation levels would give us an insight as to their current thermal sensations.

Substantial work has been done in modeling thermal comfort and making recommendations for HVAC settings in indoor environments. Most notably, Fanger [18] established the relation between predicted mean vote (PMV) and the thermal load on the body. The PMV is the mean response of a large group of people according to the ASHRAE 7-point thermal sensation scale. It was later adapted by Gao *et al.* [24, 25] to the Predicted Personal Vote (PPV) scale which is simply a personalized version of the PMV scale (applied to one person, as opposed to a group of people). Table 4.1 shows the PPV (same as the PMV) scale from very hot to very cold environments.

We adopted a divide-and-conquer approach to our problem: we divide the problem into simple sub-problems and solve them. In this work, we solve the problem

PPV	Sensation
+3	Hot
+2	Warm
+1	Slightly warm
0	Neutral
-1	Slightly cool
-2	Cool
-3	Cold

Table 4.1: PPV Index

of clothing insulation detection under different thermal comfort levels in a single occupancy setting (like vehicle interiors and office cubicles with personal heaters), we leave the multiple occupancy problem for future works. Our work is motivated by two observations. First, despite abundance of sensors that can be used to measure physical variables of the ambient environment, there is little means to measure personal variables such as clothing insulation and activity level that contribute to thermal sensation in a low-cost and non-intrusive manner. The ability to determine the latter not only allows better assessment of thermal comfort but also help identify opportunities for energy savings. Second, most existing work relies on conducting survey or soliciting explicit feedback from occupants to obtain their thermal comfort. Such a process can be intrusive and is not suitable for large-scale and/or in-situ data collection. To address these deficiencies, we devise SiCILIA, a low-cost sensor platform for occupant clothing insulation inference that aims to extract both physical and personal variables. SiCILIA consists of two components: a hardware platform that integrates multiple sensors, two servo motors, and a micro-controller, and a second component that consists of an inference algorithm to determine clothing insulation. The implementation was tested in a 2012 GMC Terrain with various passenger heating and cooling capabilities, including: heated seats, vent heating, and footwell heating. We utilize the PPV scale proposed by Gao et al. [25] as an indicator of the thermal sensation that subjects are experiencing. The traditional PMV scale is aimed at estimating the thermal comfort of a group of people. However, using the PPV scale enables us to evaluate the thermal sensations on a more granular scale, specifically on a personal scale for each individual. We demonstrate that our proof-of-concept implementation of SiCILIA can predict clothing insulations in real time. Moreover, it is capable of achieving this without the need for any training data, given properly calibrated parameters such as emissivity. The rest of this chapter is organized as follows. Section 4.1 presents an overview of clothing insulation. Section 4.3 presents the hardware platform design of SiCILIA. The clothing insulation and thermal sensation inference algorithms are then presented in Section 4.4. Section 4.5 presents the typical work flow of SiCILIA. The experimentation and evaluation process is explained in Section 4.7. Furthermore, we present an energy saving case study in Section 4.8, and finally the related works and conclusion are discussed in Sections 4.2 and 4.9, respectively.

4.1 Background and Motivation

4.1.1 Clothing Insulation

Since the early 1960s there has been a significant drive in research on thermal insulation in textile materials to find the best way to maximize thermal comfort for the wearer [19, 59, 66]. More recent efforts have focused on accommodating the occupant's environment as opposed to their clothing [4, 24, 25, 26]. Clothing insulation is the thermal insulation provided by clothing. It is typically expressed in *clo* units, where 1clo $= 0.155K \cdot m^2/W$. This is the amount of insulation that allows a person at rest to maintain thermal equilibrium in an environment at 21^oC in a normally ventilated room (0.1 m/s air movement). Typical insulation levels for clothing ensembles are given by Table 4.2 from ASHRAE [32].

Table 4.2: Thermal insulation of different clothing levels.

Daily Wear Clothing	I_{cl}
Walking shorts,short-sleeved shirt	0.36
Trousers, short-sleeved shirt	0.57
Trousers, long-sleeved shirt	0.61
Same as above, plus suit jacket	0.96
Same as above, plus vest and T-shirt	1.14
Trousers, long-sleeved shirt, long-sleeved sweater, T-shirt	1.01
Same as above, plus suit jacket and long underwear bottoms	1.30
Sweat pants, sweat shirt	0.74
Knee-length skirt, short-sleeved shirt, panty hose, sandals	0.54
Knee-length skirt, long-sleeved shirt, full slip, panty hose	0.67
Long-sleeved coveralls, T-shirt	0.72

Due to the variability of possible clothing material and type worn by anyone at any given time, the table serves only as a guide. The actual value is the numeric I_{cl} value as it is the more accurate representation of clothing thickness and insulation. For example, it is impossible to determine the clothing “shirt and undershirt” from I_{cl}, but the value itself does account for the insulation of a shirt and an undershirt.

4.2 Related Works

There is a vast amount of literature on thermal comfort and energy efficiency in building management. Most works take either an evaluative stance or a technical stance. The former being investigations of heat transfer models and factors contributing to thermal comfort, and the latter being energy management solutions that account for personalized thermal comfort.

Among the earliest works, Farnworth *et al.* presented a theoretical model that accounts for the experimentally observed heat transfer through synthetic fiber battings and a down and feather mixture in terms of a combination of conductive and radiative heat flows [19]. They have found that no convective heat flow is evident. Later, McCullough *et al.* expanded the garment and ensemble insulation values database to include more clothing ensembles and developed new methods for estimating clothing insulation. Havenith *et al.* explored the individual and combined effects of sitting, walking at two speeds, and three wind speeds on the insulation value of three clothing ensembles on: two subjects with loose fitting clothes and two subjects with tight fitting clothes [34]. Their experiments show that posture, movement, wind speed, and clothing fit all have an effect on clothing insulation. Furthermore, a linear model was proposed to estimate true clothing insulation given the different conditions under which the data was collected. However due to the myriad of different clothing styles as well as the constantly evolving nature of clothing fashion, linear models of clothing insulation need constant maintenance to keep up.

Ogulata *et al.* discussed the theoretical basis of the physical interactions between the human body's mechanism for heat transfer and it's environment. They report that in a steady state situation, the heat produced by the body is balanced by the heat lost to the environment. If not, then the occupant will experience thermal discomfort as a result of the increase in body heat temperature [66]. Zhang investigated the subjective perception of thermal sensation in individual parts of the body, and how it correlates with physiological parameters [112]. Zhang further developed predictive models of global and local thermal sensation and comfort for 19 body parts in 109 subjects. Among the main findings of the experiments, Zhang has shown that in stable conditions, local sensation has a high correlation with local skin temperature. Oliviera *et al.* presented a comparative analysis of global, serial, and parallel clothing insulation calculation methods performed on thermal manikins [67]. The global method, developed previously by Oliveira *et al.*, performs an overall calculation by summing the area-weighted of all heat losses and skin temperatures of each body part before the insulation is calculated, as if the whole manikin is made of one segment [67]. The global method serves as a general formula, while the Serial and Parallel methods are special cases. The serial method is best used with uniform insulation while the parallel with non-uniform insulation [67]. The calculation method utilized by SiCILIA is similar to the global method proposed by Oliviera *et al.*, but differs in that it also takes into account the emissivity of clothing. As a result, if the emissivity of clothing is known or calculated apriori, it can be plugged into SiCILIA's formula for a more accurate prediction. More recently, Si *et al.* presented a new methodology for the assessment of indoor comfort using environmental as well as the reported personal feeling of occupants. Objective assessment of the environment and subjective assessment of the personal feelings of the occupants are calculated and standardized in a unified scale by fuzzy fusion and adaptive fusion [94]. This allows for the classification of indoor comfort levels. In contrast to SiCILIA, this model is an active solution that relies on occupants actively reporting their thermal sensation, whereas SiCILIA works in the background without input from the user, thereby making it a passive solution.

Tackling the problem of thermal comfort assessment using clothing insulation, Gao *et al.* introduced a Smart Personalized Office Thermal Control System (SPOT). Specifically, SPOT utilizes a model called the Predicted Personal Vote (PPV), which is an extension of the Predicted Mean Vote model that allows per-user personalization. Using on-board sensors to measure various environmental parameters believed to contribute to human comfort, a linear regression model is utilized to estimate the level of clothing insulation. SPOT and SiCILIA are rather similar in purpose, but different in function. First, SPOT utilizes a Microsoft Kinect sensor for tracking purposes. While this achieves good tracking results, it is difficult to justify an "always on" camera in a home or a cubicle due to privacy concerns. Instead, we opted for a URF and the Grid-EYE infrared sensor to achieve tracking. Further, SPOT utilizes linear regression and training data from different users to infer clothing insulation. Due to the assumptions that linear regression makes (such as weak exogeneity and error independence) and the variability of human clothing fashion, it is difficult to create a regressor trained with a small sample size (20 volunteers) that is representative of the whole population and their different clothing styles. Moreover, due to the computational and power requirements of the Kinect sensor, SPOT requires a laptop in order to operate, making it less portable. SiCILIA can be used as a standalone unit as all computations are done on the on-board microcontroller. Finally, each SPOT unit costs \$1000.00 [24, 25], compared to SiCILIA which costs \$165.00 to make. Table 4.3 highlights the components of SiCILIA and the approximate cost of each one (in USD).

Gao *et al.* later introduced SPOT+ [24], which performs *predictive* control as opposed to the *reactive* control that the previous iteration [25] used. SPOT+ utilizes

Table 4.3: SiCILIA Cost

Item	**Model/Supplier**	**Quantity**	**Cost**
Grid-EYE Sensor	Panasonic	1	\$33.00
Infrared Sensor	Melexis	1	\$30.00
Arduino Uno	Arduino	1	\$30.36
Servo Motors	Parallax	2	\$39.52
Ultrasonic Range Finder	iTead Studio	1	\$3.50
Arduino LCD Display	Arduino	1	\$28.84
Total			\$165.22

pattern recognition to predict room occupancy, and learning-based model predictive control (LBMPC) to predict future room temperature.

There have also been proposals at creating more thermally aware environments for groups of people. Lam *et al.* developed an ambient temperature optimization algorithm to cater to the comfort requirements for multiple occupants [54, 69]. It is an occupant-focused approach, where the idea is to "move" an occupant's thermal environment with them as they move from room to room in a building, where the HVAC system will adjust the temperature of the room according to the thermal preferences set by the occupant. While they reported high accuracy, such approaches may suffer from setbacks before even being implemented: like incentivising occupants to provide feedback to the system. Some occupants may not adapt to the new technology and not provide feedback, as a result they may become thermally uncomfortable unless they begin to provide more feedback.

The works mentioned so far have all been aimed at uniform thermal environments. In other words, the ambient temperature has been somewhat consistent all around the occupant. In 2003, Zhang developed predictive models of local and thermal sensation and comfort for transient and non-uniform environments [112]. Interestingly, in addition to testing on 109 subjects in the UC Berkeley controlled environmental chamber, a separate set of tests in an automobile to validate their test chamber models, simulating conditions found in vehicles during both hot and cold weather [112]. The subjects body temperatures, thermal sensation and thermal comfort were measured similarly to those in the UC Berkeley controlled environmental chamber tests [112]. The validation results of using the model in the non-transient vehicular environment show that the proposed local sensation prediction model predicts the ground truth votes of subjects well.

4.3 Hardware Platform Design

The SiCILIA hardware platform has three main components: the controller, the sensor, and the motor. Figure 4.1 shows the current setup on a 2-axis skeleton design.

4.3.1 Controller

SiCILIA is driven primarily by an Arduino microcontroller. Its main purpose is sensor operation, model computation, and motor movement requests. Although a USB connection can power the Arduino and the motors, the motors would be underpowered and their sweeping speed would be significantly slower. Further, the rest of the sensors (the URF, active IR sensor and passive IR sensor array) would also be underpowered and may not report correct data (if any at all). To deal with this problem, a battery pack was added as a secondary source of energy. This improves power distribution by effectively devoting the battery pack to only the motor and the Arduino Uno, and allowing us to use the USB serial bus to send commands

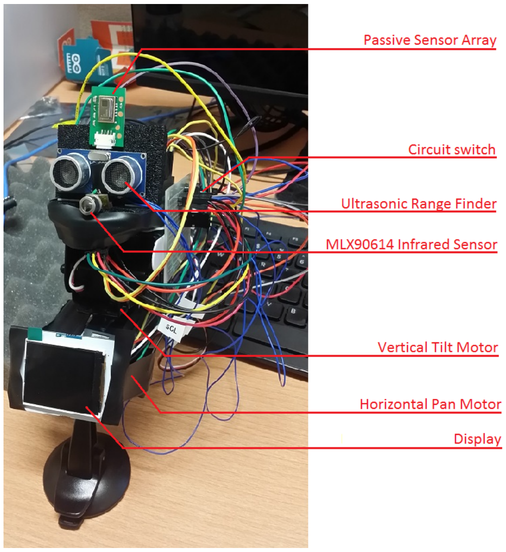


Figure 4.1: SiCILIA hardware setup.

to the system from a separate computer. However, the final version of SiCILIA is autonomous and is timer-based, and does not require any user input for operation. Additionally, extensions can be made to the Arduino Uno (such as a Wi-Fi shield) to send commands and retrieve readings wirelessly.

4.3.2 Sensors

SiCILIA uses a single MLX90614 infrared sensor to read the ambient temperature and target temperatures (i.e. clothing surface and skin). The sensor detects background thermal radiation temperature between −40°C and 125°C, and target radiation temperature between −70°C and 380°C, with a resolution of 0.02°C.

However, the MLX90614 is simply a target IR sensor and is not enough to detect movement effectively as a standalone unit. As such, SiCILIA was fitted with a Panasonic Grid-EYE thermopile IR sensor array. It contains 64 passive IR sensors arranged in an 8×8 grid fashion, each detecting the temperature of the object in front of them. The result is a heat maps of the different sources of heat in front of the sensor. Figure 4.2 shows the visual output of the array. It can be used for tracking by following the highest source of heat detected. The tracking methodology is described in Section 4.5.2.

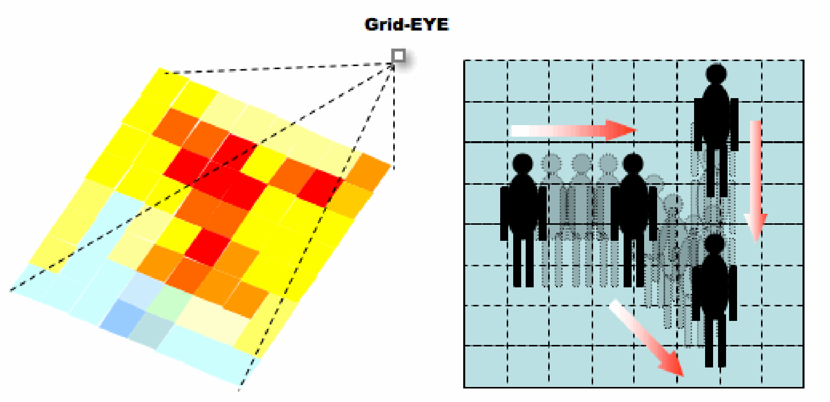


Figure 4.2: Tracking using the generated grid by Grid-EYE. [70]

SiCILIA is also fitted with an ultrasonic range finder (URF). The URF utilizes the propagation of high-frequency sound waves to find the distance between it and a remote object. Due to the diffusion of infrared heat, the further away the IR sensor is, the lower the temperature sensed will be. The distance measurement is then used to compensate for the diffusion of heat and obtain the *actual* temperature emitted from the object through an empirical model.

4.3.3 Motors

SiCILIA utilizes two continuous rotation servo motors. The motors are capable of clockwise and counter clockwise rotations. The horizontally mounted motor (hereby referred to as the x-motor) and the vertically mounted motor (referred to as the y-motor) are used in tandem to track the individual and keep the sensors aimed at the hottest area of their body. The y-motor is further used to do a vertical sweep from the individual's head to their torso to collect the relevant data needed to make an inference about the their clothing insulation.

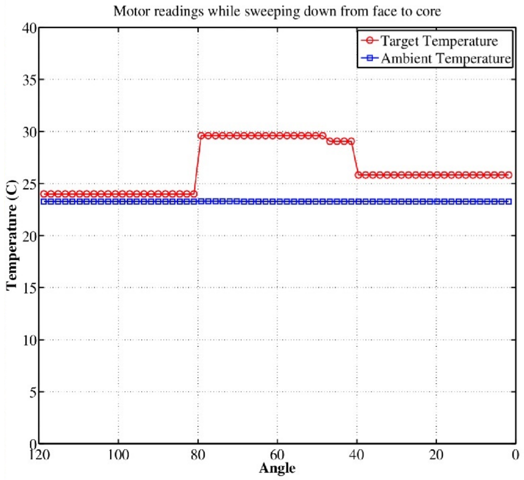


Figure 4.3: The resulting curve from the motor sweeping down from above the occupant's head to their core.

Figure 4.3 shows the readings as it sweeps from above the occupant's head to their core. Note that at the beginning, the motor is set at a high angle and cannot see the occupant, and thus the target temperature is close to the ambient temperature. SiCILIA uses a tracking algorithm to ensure the IR sensor is always looking at the occupant, hence it can assume they will always be in its Field of View (FoV). However, one cannot always assume SiCILIA is aiming at the same spot on the occupant's body every time it is used (i.e. their face, torso, etc.). Typically, the ambient temperature is lower than facial skin and clothing surface temperatures. In turn, the clothing surface temperature is lower than facial skin temperature. Therefore, sweeping can effectively determine the area of interest. Section 4.5 describes this process in more detail.

The motors are also utilized to track the occupant's movements using the Grid-EYE sensor, the algorithm is further discussed in Section 4.5. In the experiment setup, continuous rotation servos were used. Hence, it is not possible to specify the *angle* to which the servos must rotate, instead one must specify the *speed* of rotation as well as the direction.

SiCILIA uses Feetech Micro Continuous Rotation Servos (FS90R), whose maximum speed is:

1. $0.12\,\text{sec}/60^\circ$ when powered with 4.8V
2. $0.10\,\text{sec}/60^\circ$ when powered with 6V

However, with Arduino Uno the supplied voltage should be **exactly 5V**. Taking a linear approximation, the following equation can be applied to find the speed S in $s/60^\circ$:

$$S = (0.12 - 0.10) \times \frac{(5.0 - 4.8)}{(4.8 - 6.0)} + 0.12 = 0.116667\,\text{sec}/60^\circ. \tag{4.1}$$

Hardware and software constraints prevent from moving the servos continuously as one would with a stepper motor. Hence the specific angle by which the motors can "jump" by must be determined in order to track the occupant's movements accordingly. Through empirical testing, it was determined that the best angle to "jump" the motors by would be 27° as it is small enough to turn the motor in the desired direction without missing the target, but large enough to turn it significantly without needing to turn it multiple times for small distances. Hence, the servo must be turned at its maximum speed during:

$$S' = S \times (\frac{27^\circ}{60^\circ}) = 0.052s. \tag{4.2}$$

Therefore, the turn time for the motors must be approximately 0.052 seconds to turn 27°. With a frame rate of approximately 10 Hz [70], the Grid-EYE keeps the motors updated on what movements are happening and directs them as to which way to turn. This is enough to reliably track humans walking at an average speed of approximately $3km/hr$ to $5km/hr$ at a maximum distance of approximately $100cm$. However, due to the nature of the servo motors and the "jumping" effect described earlier, it becomes rather difficult to track objects moving at faster speeds. This problem can be resolved by upgrading to stepper motors.

Another dimension that must be tackled with tracking is the distance away from the Grid-EYE sensor. If an occupant is walking at $5km/hr$ at a distance of $100cm$ relative to the sensor, they will spend *more* time in the FoV of the sensor than if they were walking at the same speed at $30cm$ away. Hence, the motor tracking movement must also be adjusted according to the distance away from the sensor. Using the URF it is possible to infer the distance, the direction and speed in which the motors must move. However we have left this problem to future work.

4.4 Inference of Clothing Insulation

During equilibrium, the result of the difference between skin temperature (or approximately, the inner surface temperature of the clothing) and the outer surface temperature (i.e. the conduction heat transfer) should be the same as the heat loss to the environment by convection and radiation. In Figure 4.4, C_k stands for the

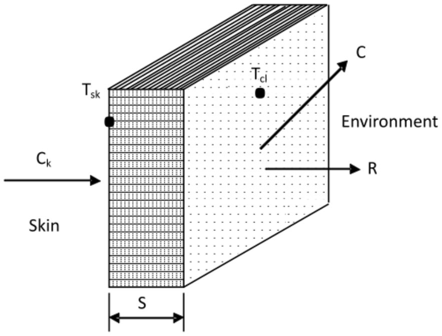


Figure 4.4: Convection, conduction, and radiation acting on clothes as highlighted by Ogulata *et al* [66].

conductive heat transfer from the inner surface to the outer surface of the clothing, and is defined as [66]:

$$C_k = (T_{sk} - T_{cl})/R_{cl}, \quad \text{where} \quad R_{cl} = 0.155 I_{cl}, \tag{4.3}$$

where T_{sk} and T_{cl} are the skin temperature and clothing surface temperature in Kelvin [2], respectively, and R_{cl} is the insulation of clothing measured in SI unit.

Typical *clo* values of clothing assemblies are given in Table 4.2. Furthermore, a few constants were assumed during experiments: first is the Stefan-Boltzmann constant, denoted as σ and valued at $5.67 \times 10^{-8} W \cdot m^2 \cdot K^4$. Secondly, we assume the emmisivity of the clothing to be close to black-body emissivity at 0.9, and is denoted by ϵ_{cl}. Finally, the air velocity to be at $0.20 m/s^2$, which is typical in indoor office environments, and is denoted by V_a.

Let C be the convective heat loss, and R be the heat loss by thermal radiation

---

[2] $[K] = [^oC] + 273.15$

from the clothing surface. In equilibrium,

$$C_k = C + R. \tag{4.4}$$

What remain to be determined are C and R. From [9, 17, 30], the convection heat loss is given by (4.5),

$$C = f_{cl} h_c (T_{cl} - T_a), \tag{4.5}$$

where f_{cl} is clothing area factor, h_c is the coefficient of convective heat transfer (W/m^2K) and T_a is the ambient temperature. f_{cl} is the ratio in surface area between the outer clothing surface and the nude person's surface area [34], given by

$$f_{cl} = 1.05 + 0.1 I_{cl}, \tag{4.6}$$

h_c is given by,

$$h_c = 12.1 V_a^{0.5}, \tag{4.7}$$

where V_a is the air velocity. V_a can be measured directly. In our derivation, we assume $V_a = 0.20 m/s$ as specified by ASHRAE handbook [31].

The heat loss through radiation can be determined by [9, 17, 30]:

$$R = \sigma \epsilon_{cl} f_{cl} F_{vf} [T_{cl}^4 - T_a^4], \tag{4.8}$$

where ϵ_{cl} is the emissivity of clothing, F_{vf} is the view factor between the body and the surrounding surface, σ is the Stefan-Boltzmann constant (σ). The emissivity of the clothing and skin is very close to that of a black body, and thus has a value of nearly 1. The effective area of the body for radiation is roughly 80 percent of the total area, and thus, $F_{vf} = 0.80$.

To this end, combining (4.3) – (4.8) creates:

$$\frac{T_{sk} - T_{cl}}{0.155 I_{cl}} = f_{cl} h_c (T_{cl} - T_a) + \sigma \epsilon_{cl} f_{cl} F_{vf} [(T_{cl} + 273.15)^4 - [(T_a + 273.15)^4]. \tag{4.9}$$

In (4.9), δ, F_{vf}, h_c are constants; T_a, T_{cl}, T_{sk} are measured via the infrared sensor. Note f_{cl} relates to I_{cl} by (4.6). Thus, I_{cl} is the root of a quadratic equation. To simplify the equation, assume $C_1 = (h_c[T_{cl} - T_a] + \sigma \epsilon_{cl} F_{vf} [(T_{cl} + 273.15)^4 - [(T_a + 273.15)^4]$ and $C_2 = T_{sk} - T_{cl}$. Hence the I_{cl} is given by:

$$I_{cl} = \frac{-0.116 C_1 \pm \sqrt{(0.116 C_1)^2 - 4 \times 0.0115 C_1 C_2}}{2 \times 0.0115 C_1}. \tag{4.10}$$

Naturally, solving this for I_{cl} will output two values for I_{cl}. The value that falls between 0 and 1.5 is the predicted clo value. However, a few comments are in order. First, the parameters used in the derivation are chosen from typical indoor settings. In cases where the actual value deviates by much, the model would not produce very accurate results – one such example being a shiny clothing material with low emissivity like polyester. Clothing with low emissivity is designed to preserve body heat, and hence it is more commonly worn outdoors. Since SiCILIA is primarily made for indoor use, the focus is not on emissivity as it is not a trivial task and is out of the scope of this project.

Second, Ogulata [66] used a similar model to find the clothing surface temperature (T_{cl}). In contrast, we solve for clothing insulation I_{cl} from measured clothing surface temperature T_{cl} and other quantities. Third, if $(0.116C_1)^2 - 4 \times 0.0115C_1C_2$ is negative, then (9) has no solution. This may happen when the ambient temperature is very high and as a result the clothing surface temperature is too close to the ambient temperature.

4.5 Work Flow

The ultimate design goal of SiCILIA is a passive-active sensor that seamlessly integrates into the user's environment. Using the hardware setup and inference algorithm described in Sections 4.1 and 4.4 respectively, we describe how SiCILIA can function in a stand-alone manner without human intervention. Initially, SiCILIA will remain in the STANDBY state (that is, simply waiting) until a heat-emitting object moves into its FoV, in which case it will transition into the READY state and begin tracking it. Simultaneously, a countdown timer is started to determine whether the subject has been stationary for long enough such that scanning (or measuring) will yield a feasible result. Once the timer expires, SiCILIA will transition to the SCANNING state and begin the measurement. Once the scanning is finished, SiCILIA will go back to the READY state and reset the timer. If the object leaves the FoV of the sensors entirely, then it will go back to the STANDBY state.

Figure 4.5 shows the work-flow in between these 3 states. We will now go into further detail about each state and its transition.

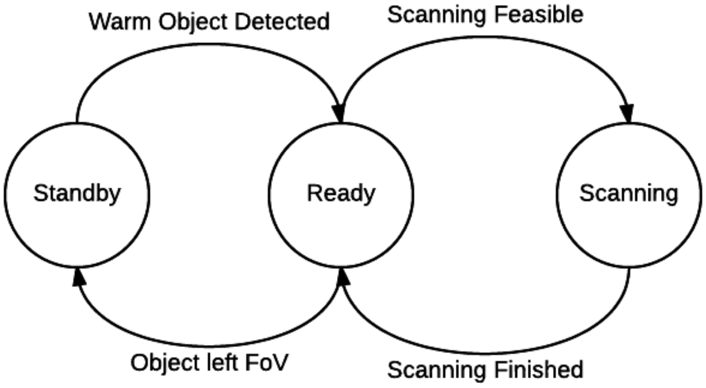


Figure 4.5: Usage procedure state diagram

4.5.1 Standby

The de-facto state of SiCILIA, it remains in this state for as long as no heat-emitting object has been detected within the FoV of the Grid-EYE sensor. It will also return to this state if a heat-emitting object has left its FoV completely.

4.5.2 Ready

SiCILIA will remain in this state for as long as there is a heat-emitting object within its FoV. In this state, SiCILIA is actively doing two tasks simultaneously:

1. Tracking the object using an algorithm based on the center of mass equation.
2. Using a timer to determine whether the object in the FoV is eligible for a clothing insulation measurement. If so, it will shift into the SCANNING state.

We will now describe (1) and (2) in further detail.

Tracking Since the Arduino Microcontroller has limited processing capabilities, a more simplistic model for tracking the occupants was chosen. SiCILIA finds the spot with the highest concentration of heat on the 8×8 grid generated by the Grid-EYE sensor and follows it by panning and tilting the sensors using the x- and y-motors, respectively. This has two benefits:

1. It allows us to track only the closest occupant in proximity to the sensor, and safely ignore any occupants standing further away from the sensor.
2. Due to the simplicity of this calculation, tracking is almost instantaneous and hence does not create any delay overheads before moving the motors.

In order to find the concentration of heat, the center of mass equation was adapted to the 8×8 grid. The center of mass is the unique point at the center of a distribution of mass in space where the weighted position vectors relative to this point add up to zero. In terms of statistics, the center of mass is the mean location of a distribution of mass in space. This adaptation, called the center of heat equation, finds the centroid of the heat grid generated by the Grid-EYE:

$$\vec{v} = (x_i, y_j), \qquad c = \frac{\sum_{i,j=0}^{n} t_{ij} \cdot v}{\sum_{i,j=0}^{n} t_{ij}}, \tag{4.11}$$

where x_i and y_j represent row i and column j of the 8×8 grid generated by the Grid-EYE sensor, respectively, and n is the number of values per row/column (8 in this case, since it is an 8×8 grid). Further, t_{ij} represents temperatures at row i and column j, respectively. The result is a vector $\vec{v_{ij}} = (x_i, y_j)$ that points to the highest concentration of heat on the grid. The horizontal and vertical motors then move the platform such that (x, y) is now at the center of the grid.

Timer The nature of the occupant's occupancy and movement relative to the sensor is unpredictable. For example, the occupant may be in the area to do a simple task and then leave immediately, or they may be in the area for hours on end. As such, SiCILIA must be certain that the occupant will not be moving significantly when SiCILIA scans their clothing insulation. To this end, SiCILIA uses a countdown timer to determine whether the occupant is stationary long enough to take a reliable measurement. Once SiCILIA enters the READY state, the timer begins counting down to 0. If the occupant moves such that SiCILIA must move its motors to place him/her in the center of the FoV of the sensors again, then the timer is re-set and started again. If the timer hits 0 without any motor movement, it is eligible for a measurement and SiCILIA will enter the SCANNING state. The timer value can vary according to the application. For example, it can be as long as 5 minutes in an office cubicle setting (since occupants will typically remain in cubicles for long periods of time), or 30 seconds in a vehicular setting (since the time spent in a car is variable and can range from minutes to hours).

4.5.3 Scanning

Once SiCILIA determines an object is indeed ready for a measurement (that is, the timer hits 0), it will shift into the SCANNING state and begin the measurement procedure. While in this state, tracking is disabled so that the MLX90614 IR sensor can take control of the motors to vertically scan (or "sweep") the occupant. Once the measurement is finished, SiCILIA will go back to the READY state. In order to use the inference algorithm described in Section 4.4, SiCILIA needs to record the skin temperature, clothing surface temperature, and the ambient temperature. The steps taken within the Scanning state to obtain these values is described below:

1. Record the ambient temperature.
2. Use the MLX90614 IR sensor to record the target temperature. If the temperature reads higher than 31°C then it is likely aimed at the occupant's skin, and will record the temperature as the skin temperature. Otherwise, tilt upwards until a temperature reading higher than 31°C is found. The reason that the threshold was set at 31°C is because human skin temperature typically rests at 32°C [51].
3. Tilt downwards until a temperature reading lower than 31°C and above the ambient temperature is found.
4. Plug in the skin, clothing surface, and ambient temperatures to (4.10).

The skin surface temperature is typically higher than the clothing surface temperature, which in turn is higher than the ambient temperature. If at any point the target temperature drops down to a value close to the ambient temperature, then it is reasonable to assume that SiCILIA is no longer pointing at the occupant. In this case, the inference from Equation (4.10) will be a negative value, and is discarded. SiCILIA will return to the Ready state regardless of the measurement outcome, and will measure again once the timer times out.

4.6 Other Design Considerations

In order for SiCILIA to become a fully automated entity, a myriad of concerns must be resolved. We have tackled a few of those in this work to create a working prototype, however other design considerations may arise as the platform evolves.

4.6.1 Angle Compensation

As shown in Figure 4.1, the sensors on-board SiCILIA are “stacked” in the sense that they are placed on top of each other. At the bottom is the active IR sensor (MLX90614), in the middle lies the URF, and at the very top rests the Grid-EYE passive sensor array. Since the Grid-EYE acts as the “eyes” of SiCILIA, the other two sensors must be aligned accordingly in order to adequately capture the skin and clothing surface temperatures. Most importantly, the IR sensor must be tilted such that it points at the same area that the Grid-EYE was pointing at earlier. The URF has a bit more “wiggle room” since the object being tracked is intended to be human, hence the distance will always remain relatively the same as long as the sensor is pointing in the same direction as the others (for example, the distance of a person’s face from the URF will be the same as their chest). Therefore, SiCILIA must ensure that the IR sensor captures the same information as the Grid-EYE sensor.

Figure 4.6 shows a typical setup between the IR sensor, the Grid-EYE, and a human’s face. There are two given facts in this problem:

1. The Grid-EYE will always be at a 90° angle relative to the MLX sensor and the face, and hence SiCILIA must determine the appropriate angle in which the IR sensor should be angled to ensure it is pointing at the same object as the grid-EYE.

2. The distance between the Grid-eye sensor and the MLX90614 sensor will always be 4.8cm due to the hardware platform structure. This value may be adjusted if SiCILIA is re-designed.

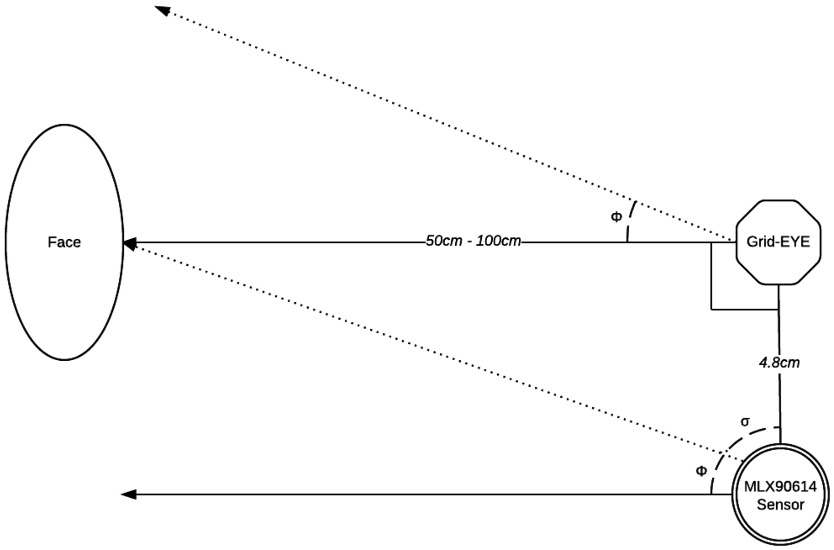


Figure 4.6: The relationship between the MLX90614, Grid-EYE sensor, and a human

Re-angling the IR sensor is tedious and difficult to do without a tertiary motor to automate the process, which in turn will increase costs of power and operation. Therefore, to tilt the IR sensor to aim at the same target that the Grid-EYE is aiming at, SiCILIA moves the entire sensor tray. The question then becomes "what is the angle at which the tray should tilt such that the IR sensor is pointed at the same object that the Grid-EYE sensor is pointing at now?".

Using some elementary geometry, it is possible to derive the angle at which the tray should tilt to aim the IR sensor at the spot where the Grid-EYE was pointing. Let σ denote the current angle of the IR sensor relative to the user's face, and Φ denote the angle in which the platform needs to be tilted such that the IR sensor will be aiming at the face. Since the IR sensor senses target temperatures as well as their distance d away in a relatively straight line in front of it, it is safe to assume that $\sigma + \Phi = 90°$. Therefore,

$$90^\circ - \sigma = \Phi, \tag{4.12}$$

meaning σ must be determined in order to find Φ. Therefore:

$$\sigma = \tan^{-1}(\frac{d}{4.8cm}). \tag{4.13}$$

Hence, the IR sensor is at an angle of σ relative to the occupant's face. Using Equation (4.12), it is clear that the sensor tray must tilt by $\Phi = 90^\circ - \sigma$ to aim the IR sensor at the occupant's face. Let us take an example where SiCILIA detects an object to be $50cm$ away from it. Hence:

$$\sigma = \tan^{-1}(\frac{50cm}{4.8cm}) = 84.5^\circ. \tag{4.14}$$

Applying Equation (4.12), it becomes clear that the system should tilt by $90^\circ - 84.5^\circ = 5.5^\circ$ upwards to make up for the difference between what the Grid-EYE and the IR sensor are pointing at. Similarly, if the object is $100cm$ away then $\Phi = 2.7^\circ$, whereas if the object is $20cm$ away then $\Phi = 13.5^\circ$. This indicates that the further away the occupant is from SiCILIA, the smaller the tilt angle becomes. This is because as the distance between the occupant and SiCILIA increases, so does the amount of the occupant's body "seen" inside SiCILIA's FoV. For example, if an occupant is $30cm$ away, then perhaps only their chest would be in the FoV, whereas if they were $200cm$ away, then their whole body would be in the FoV. However, as the distance increases, then their "image" in the IR sensor becomes smaller and the perceived target temperature becomes less accurate. The issue of distance compensation is tackled in Section 4.6.2. Additionally, it is possible to create a "bound angle" for both approaches that the platform will abide by in order to limit the time and the distance of the sweep. The bound angle will simply limit the vertical sweep such that the platform will not look too far up or too far down trying to find an occupant's face or torso.

4.6.2 Distance-base compensation

The intensity of light or other linear waves radiating from a point source is inversely proportional to the square of the distance from the source [37]; so an object of the same size that is twice as far away receives only one-quarter the energy in the same time interval. That is the inverse-square law in terms of electromagnetic radiation.

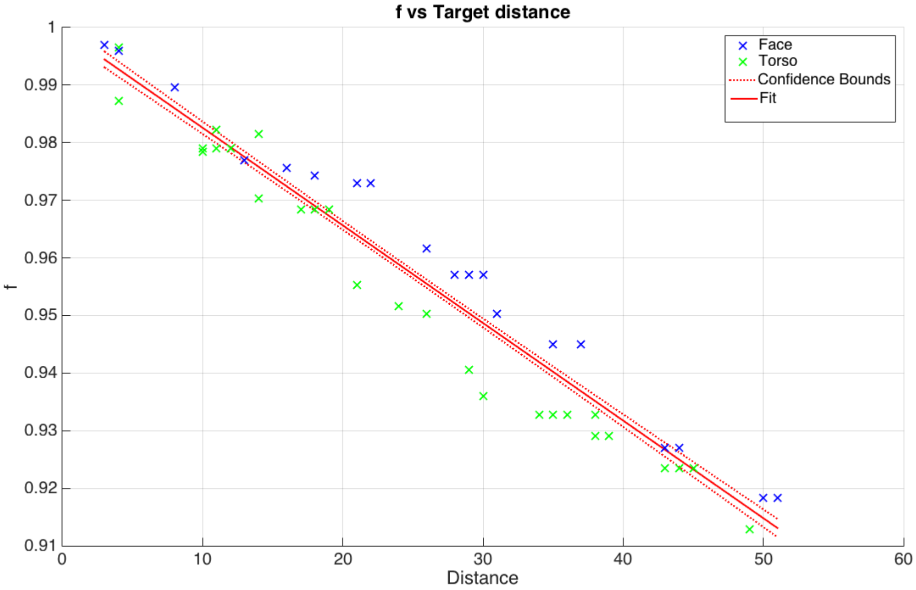


Figure 4.7: Ratio between the true object temperature and the perceived temperature of the target (face and torso) vs the distance

Further, the issue with distance and IR sensors such as the MLX90614 that SiCILIA utilizes is that the *image* of the object being measured gets smaller and smaller as the distance increases, and soon the temperature of the object will be averaging out with other objects around it and the ambient temperature. Consequently, the distance between the user and SiCILIA should be used to compensate for the perceived temperature every time a measurement is taken, since the user may have shifted in their seat or moved in between measurements. To this end, we train a linear model using the heat temperature data of the torso and the face of 10 participants at different distances between 5 cm and 50 cm. $f = \frac{t_d}{t_o}$ is defined as the ratio between the original (t_o) and the perceived (t_d) temperatures (measured in degrees Celsius) at a distance. This ratio is used as the "amplifier" to the perceived temperature t_d to find the true object temperature t_o using the following equation:

$$\hat{t}_o = \frac{t_d}{f}. \tag{4.15}$$

Figure 4.7 shows the relationship between f and the distance from the face and torso, respectively. To find the true temperature of an object, a classifier was built in order to predict what the original temperature of an object is given the distance and the perceived temperature of the object at that distance. To create a good classifier, it is essential not to over-fit or under-fit the data. Hence linear regression with gradient descent was used to minimize the cost function and find the curve of best fit.

$$f = 0.99954 - 0.0017 \cdot d. \tag{4.16}$$

Once the amplifier value f becomes known for face and torso temperature amplification, SiCILIA plugs the values into f in Equation (4.15) to obtain the actual temperature of the target. We used k-fold cross validation to check the performance

Table 4.4: Comparison between t_o/t_d and regressor f

Distance	t_o	t_d	f	$\hat{t}_o = t_d/f$
5 (cm)	32.00°C	31.89°C	0.991	32.21°C
10 (cm)	32.00°C	31.33°C	0.982	31.90°C
15 (cm)	32.00°C	31.05°C	0.974	32.00°C
20 (cm)	32.00°C	30.99°C	0.967	32.04°C
25 (cm)	32.00°C	30.45°C	0.957	31.81°C
30 (cm)	32.00°C	30.10°C	0.949	31.72°C
35 (cm)	32.00°C	29.85°C	0.940	31.75°C
40 (cm)	32.00°C	29.73°C	0.931	31.93°C
45 (cm)	32.00°C	29.55°C	0.923	32.02°C
50 (cm)	32.00°C	29.21°C	0.914	31.96°C

of the model by partitioning the dataset into 100 partitions of 23 data points (out of 230 data points), and dividing them into training and test data. The average root mean square error (RMSE) was found to be approximately 0.0056. To further verify the validity of our regressor, we have performed an experiment in which an occupant's face temperature was measured in contact with SiCILIA's IR sensor and then measured it again at distances ranging from 5 cm to 50 cm. The perceived object temperature $\hat{t}_o$, defined in (4.15) is compared to the true object temperature t_o in Table 4.4. Indeed, $\hat{t}_o$ is always close or equal to the true object temperature t_o. This confirms that f is reliable enough to be used to compensate for temperature at a distance.

4.7 Experimental Validation

To validate the design of SiCILIA, there are two functions to evaluate: tracking, and clothing inference. In the case of tracking, experiments were run on one subject moving in different directions. For clothing insulation, experiments were run on a total of ten subjects (5 male, 5 female), each with three different levels of clothing insulation. For each level, the clo value was determined based on the garment insulation table in [32]. Table 4.5 shows what each subject was asked to wear for each clothing insulation value during the experiments.

Table 4.5: Clothing worn by subjects for each insulation value during experiments

	0.45 clo	**0.70 clo**	**0.81 clo**
Underwear (cotton) **(0.04 clo)**	✓	✓	✓
Short sleeve cotton T-shirt **(0.17 clo)**	✓	✓	✓
Denim jeans **(0.24 clo)**	✓	✓	✓
Long sleeve sweater(thin) **(0.25 clo)**	▨	✓	✓
Long sleeve sweater(hoodie) **(0.36 clo)**	▨	▨	✓

Subjects are of different ages (from 20 to 60 years old), body weights (110 lbs. to 282 lbs.), and Body Mass Indices (BMI) (from 19.5 (normal weight) to 36.2 (obese)). The full subject data can be found in the Appendix. We first present the evaluation of the tracking algorithm, and then present the evaluation of the clothing inference algorithm.

4.7.1 Occupant Tracking

Since SiCILIA requires tracking the user before performing a measurement, it is imperative to choose a robust tracking algorithm. Using a camera or a Kinect sensor has been utilized by Gao et al. [24, 25] with high accuracy. However, an always-on camera and the computational requirements of the Kinect introduce privacy and computational overhead problems, respectively. Instead, SiCILIA's on-board pyroelectric infrared sensors are a good approach to achieve good tracking performance, resolve privacy concerns, and maintain a low computational overhead. Additionally, the problem of human tracking and localization using pyroelectric infrared sensors has

Table 4.6: Confusion matrix for occupant half meter (50cm) away

Prediction \ Ground Truth	Left	Right	Middle	Up	Down
Left	90%	0%	10%	0%	0%
Right	0%	92%	4%	4%	0%
Middle	6%	4%	90%	0%	0%
Up	0%	2%	10%	88%	0%
Down	2%	0%	6%	0%	92%

been investigated in the past [35, 65, 96]. However, these works implement tracking for surveillance and localization purposes and as such employ sophisticated solutions. SiCILIA is only required to track objects that are:

1. In the FoV of the Grid-EYE sensor.
2. A maximum distance of $100cm$ away from it.
3. Are going to remain in front of it long enough to take a measurement.

These three conditions simplify the tracking requirements. It becomes possible to create a tracking algorithm that is robust enough to keep the user in the center of SiCILIA's FoV at all times and computationally simple enough to run on a microcontroller. the algorithm must be capable of differentiating between the different directions that a subject may take at different distances. Two distances (50cm and 100cm) and four directions of movement were evaluated: left, right up, and down. Additionally, the motor's reaction when the subject is already directly in front of SiCILIA (marked as "middle" in the confusion matrices) was also recorded. The subject moved 50 times in each direction and the motor's reaction was recorded. Tables 4.6 and 4.7 show the confusion matrices generated from this experiment.

Table 4.8 shows the precision and recall rates for $50cm$ and $100cm$ cases. Note how they both deteriorate as the distance increases. This is due to the occupant's body

Table 4.7: Confusion matrix for occupant one meter (100cm) away

		Ground Truth				
		Left	Right	Middle	Up	Down
Prediction	Left	80%	0%	10%	10%	0%
	Right	0%	84%	10%	4%	2%
	Middle	6%	8%	86%	0%	0%
	Up	0%	4%	20%	72%	4%
	Down	2%	4%	30%	0%	64%

Table 4.8: Precision and Recall rates for 50cm and 100cm cases

	Precision (50cm)	Recall (50cm)	Precision (100cm)	Recall (100cm)
Left	0.90	0.92	0.80	0.91
Right	0.92	0.94	0.84	0.84
Middle	0.90	0.75	0.86	0.55
Up	0.88	0.96	0.72	0.84
Down	0.92	0.92	0.64	0.91

occupying less of the FoV of the Grid-EYE sensor and hence less heat is perceived. If distance compensation on the Grid-EYE is applied, then it may be possible to retrieve the true temperature for each point on the 8×8 grid. However, this opens other dimensions to the problem as Grid-EYE's view must now be treated as a 3D figure since SiCILIA first has to extract the user's body and then discard the background heat behind it. Hence, we have left the problem to be solved in future works.

Furthermore, in the case of the user moving up and down the precision falls significantly when the distance increases. This is also because more of the user's body becomes inside the FoV of the Grid-EYE as the distance increases. Specifically, as more of the user's body goes into the FoV, the sensor stops seeing individual concentrations of heat on their body and begins to see the entire body as *one* point of concentrated heat. Consequently it becomes difficult to differentiate between standing up and sitting down, and the algorithm begins to infer the user as simply standing still in the middle of the FoV.

Tracking Timing

Part of the problem of tracking occupants is *when* to track them. As mentioned previously, SiCILIA's tracking algorithm places the occupant in the center of the GridEYE's vision. This is important as a user may be in the vicinity for only a short amount of time, and if SiCILIA takes too long to track and position itself correctly then it may miss a scheduled measurement. We have evaluated this metric at two distances and two clo values: 50cm and 100cm, and 0.45 clo and 0.81 clo, respectively in table 4.9. The reason for using different clo values is because the more insulation that a person is wearing, the less heat will be detected by the GridEYE sensor. However, the difference in tracking timing was not significant as SiCILIA is usually able to lock on to the user's face which was always uncovered in the experiments. As such, we conclude that as long as SiCILIA "sees" a person's face, then it will be able to lock onto them within one second.

Table 4.9: Time SiCILIA takes to find occupants

	0.45 clo (50cm)	0.81 clo (50cm)	0.45 clo (100cm)	0.81 clo(100cm)
Time	0.76 sec	0.77 sec	0.84 sec	0.89 sec

Tracking Consistency

In addition to quick tracking, it is important to have *consistent* tracking. This means that SiCILIA must be tracking the occupant correctly for as long as it is in use. We determine "correct tracking" by the way SiCILIA positions itself relative to the occupant: if the sensors are aimed at the occupant's face or torso, then the tracking is correct and SiCILIA is ready to take a measurement at any time. Otherwise the tracking is determined to be “incorrect”. To determine how consistent SiCILIA is, we ran an experiment for an occupant using SiCILIA for one hour. SiCILIA was placed on a cubicle desk and tracked the occupant for 60 minutes. During those 60 minutes, the occupant would perform usual daily tasks at their desk, like:

1. Typing on the computer.
2. Writing on a piece of paper.
3. Leaving the cubicle and coming back.
4. Stretching.
5. Turning around and speaking to the person in the adjacent cubicle.

SiCILIA tracked the occupant correctly for 93% of the time. During the tracking period, two things occurred that “muddled” SiCILIA’s tracking by 7%:

1. Another person entered SiCILIA’s FoV and it was unable to determine which person to track during this time. Once the person left, SiCILIA went back to tracking the occupant sitting at the desk.
2. At some point, the occupant was drinking a hot cup of coffee. When the person would hold the cup, SiCILIA would focus on the cup and not the person holding it. This is due to the cup being hotter than the surface temperature of the person. Once the cup was placed away from SiCILIA’s FoV, it went back to tracking normally.

4.7.2 Test procedure

The system was tested inside a 2012 GMC Terrain equipped with temperature-controlled air conditioning (AC), heated seats, vent heating, and footwell heating. However, most of these features may affect certain body parts more than others, which may skew or bias the data as a result. Since our aim is to affect as much of the occupant's body as possible all at once, we opted to use windshield vent heating so as not to blow at any part of the occupant directly. Rather, it blows into the ambient air in the vehicle, heating and cooling it slowly. Even with these precautions, vehicle thermal environments remain transient and the temperatures are non-uniform as compared to typical office cubicles which are relatively non-transient in nature [82] due to the heating and cooling air not being directly pumped at occupants. Consequently, we reason that testing our implementation in a vehicular environment provides for a more challenging environment for SiCILIA, confirming that it is functional both in a vehicular setting as well as a less transient setting such as a small room or a cubicle.

In order to obtain a variety of PPVs from each subject (i.e. from $PPV = -3$ to $PPV = +3$), the AC temperatures were adjusted from "very cold" to "very hot" (we do not specify temperatures here as each subject specified different temperatures in the spectrum of "very cold" to "very hot"), with the air coming out of the windshield vents to provide for a more uniform temperature experience. Each experiment proceeded as follows:

1. Set temperature to comfortable level for the occupant, where $PPV = 0$
2. Wait 15 minutes for the occupant's body to reach thermal equilibrium, then take clothing insulation measurements.

3. Gradually decrement temperature by 0.5°C until user votes $PPV = -1$ (slightly cool).

4. Repeat step 2 and 3 for $PPV = -2$ and $PPV = -3$.

5. Bring occupant back to normal comfort level ($PPV = 0$).

6. Gradually increment temperature by 0.5 °C until user votes $PPV = +1$ (slightly warm).

7. Repeat step 2 and 6 for $PPV = +2$ and $PPV = +3$.

This procedure was repeated 3 times: once for each clothing insulation value in Table 4.5. To test the proposed model, each subject was asked to wear the clothing specified in Table 4.5. Their clothing was then evaluated using SiCILIA, which was placed on the vehicle dashboard in between the driver and the passenger. The test subject sat in the passenger seat and the examiner in the driver's seat. SiCILIA first makes a vertical sweeping motion, initially pointing at the occupants face and to measure the skin's surface temperature, and then sweeps down the occupant's core and measuring the average surface temperature of their clothing.

4.7.3 Results

Figure 4.8 shows the inferred clothing insulation from (4.10) for all subjects under different PPVs (starting with $PPV = -3$ to $PPV = +3$) in phase 1 of the experimentation. Note that in the case of $PPV = +3$, subjects 3, 4, 7, and 8 do not have a reading as they did not vote any higher than $PPV = +2$ for the temperatures where the rest of the subjects voted $PPV = +3$. This leads to the conclusion that while all subjects will report an "extremely cold" sensation ($PPV = -3$), *some* subjects will not be as uncomfortable in $PPV = +3$, this could be due to the thermal background of the subjects upbringing. Meaning subjects that grew up in hot or cold areas may be more resistant to hot and cold temperature than other subjects. However, the specific cause of this is beyond the scope of this work. From Figure 4.8, it is evident the inferred *clo* values are very close to the ground truth, with an RMSE of ±0.08. This is comparable with SPOTs RMSE of 0.0919 [25]. However SiCILIA differs on its portability, low computational overhead, and immediacy of use as it does not require training data for clothing insulation prediction. Figure 4.9 shows the curves $PPVs$ as the subject's clothing insulation changes, the similarity of these curves shows that SiCILIA can infer clothing insulations at all subject comfort levels, and at different temperatures ranging from 14°C to 35°C. Hence we demonstrate that SiCILIA can provide reasonable results as long as the type of clothing worn is reasonably emissive of the wearer's body heat, insulated clothing tends to have a similar surface temperature to the ambient temperature, skewing results.

Physiological effects

For each subject we recorded gender, age, height, and weight. This information is used for the calculation of the Body Mass Index (BMI). Derived from the weight and height of an individual, BMI quantifies the amount of tissue mass (muscle, fat, and bone) in an individual, and then categorize that person as underweight, normal weight, overweight, or obese based on that value [2]. Figure 4.10 shows the relationship between BMI and the typical PPV vote from a subject with a specific BMI. Indeed, subjects with a low BMI tended to feel, on average, colder than subjects with a

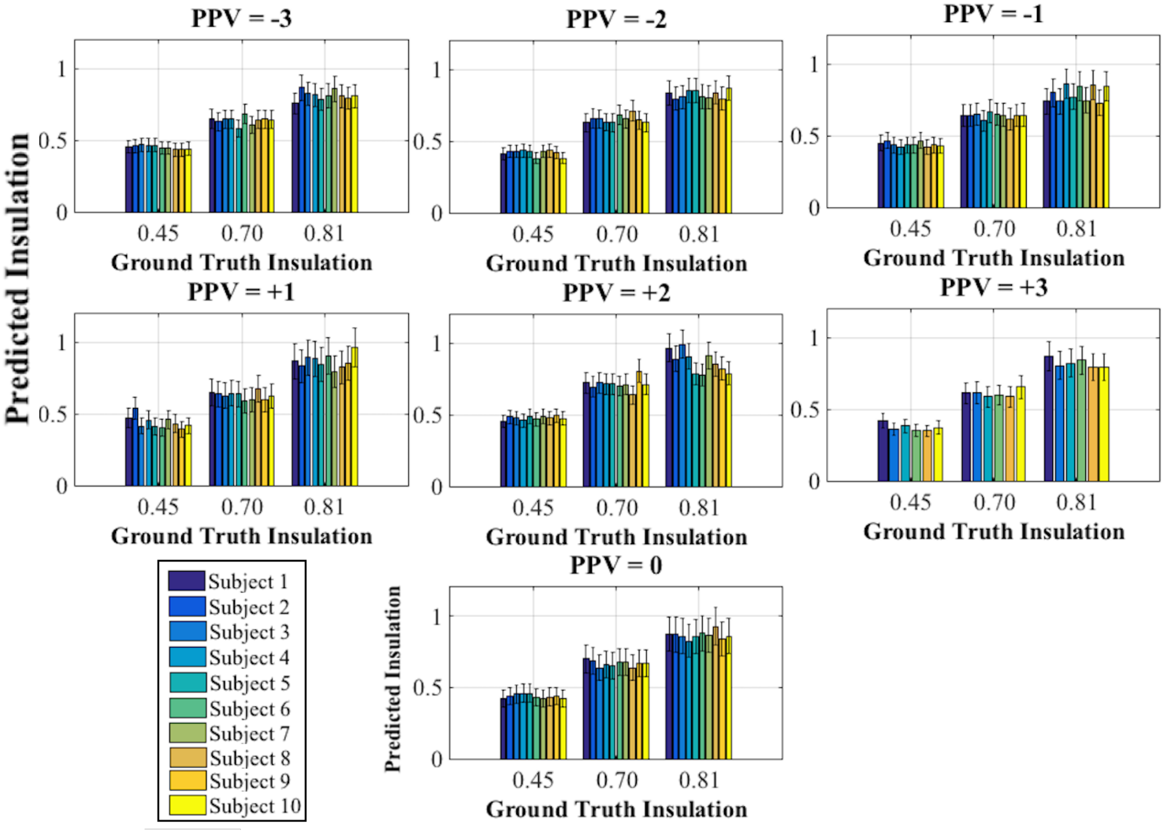


Figure 4.8: Subject clothing insulation predictions in $PPV = -3$ to $PPV = +3$

higher BMI. This can be due to the absence of fatty tissue (which provides some level of insulation) compared to subjects with a high BMI, who tend to have more fatty tissue. The accuracy of BMI is disputed among the medical community [105, 100], and the statistical community [53] due to its limitations regarding height scaling and indifference to muscle and fat masses. Therefore, this trend should serve as simply a guideline since the sample size is not large and BMI itself is not an accurate measure of body composition in an individual.

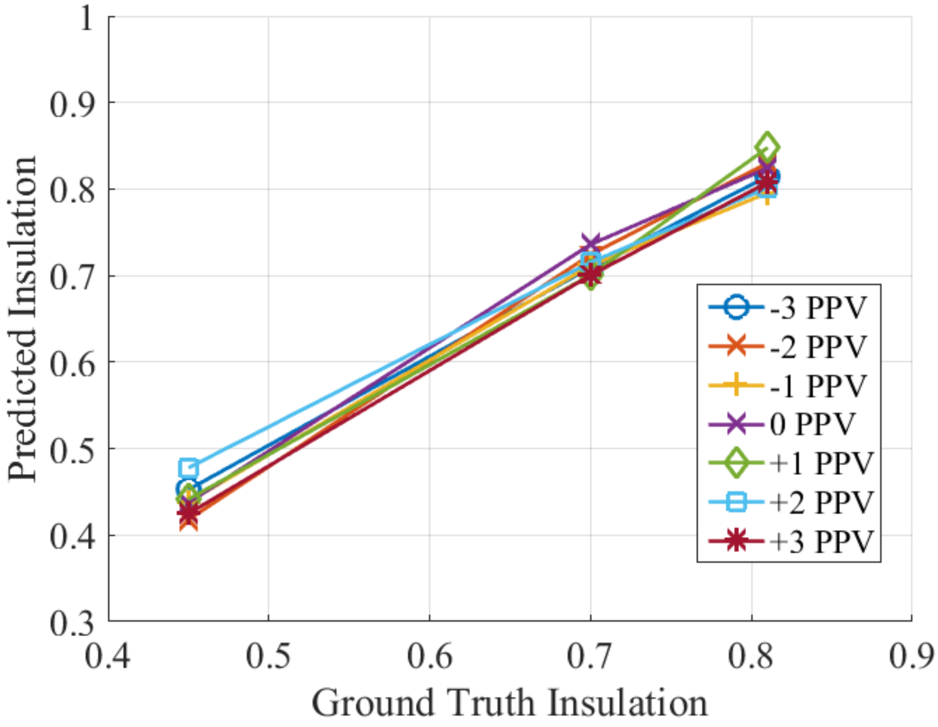


Figure 4.9: Clothing insulation predictions in changing PPVs and insulation values

4.8 Case study

We present a back-of-envelop study on the potential savings in energy consumption with and without clothing insulation inference in an office building. The range of temperatures and humidity under specific clothing insulations in which an occupant will be thermally comfortable are characterized by *comfort zones*. In order to calculate the user's thermal comfort zone, we have utilized an online tool developed by the Center for Built Environment (CBE) at the University of California at Berkeley [38]. The tool follows the ASHRAE standards [31] in calculating the user's thermal comfort relative to their thermal comfort zone, given inputs such as the air temperature, the clothing insulation level, air speed, and humidity. For the purposes of this study we

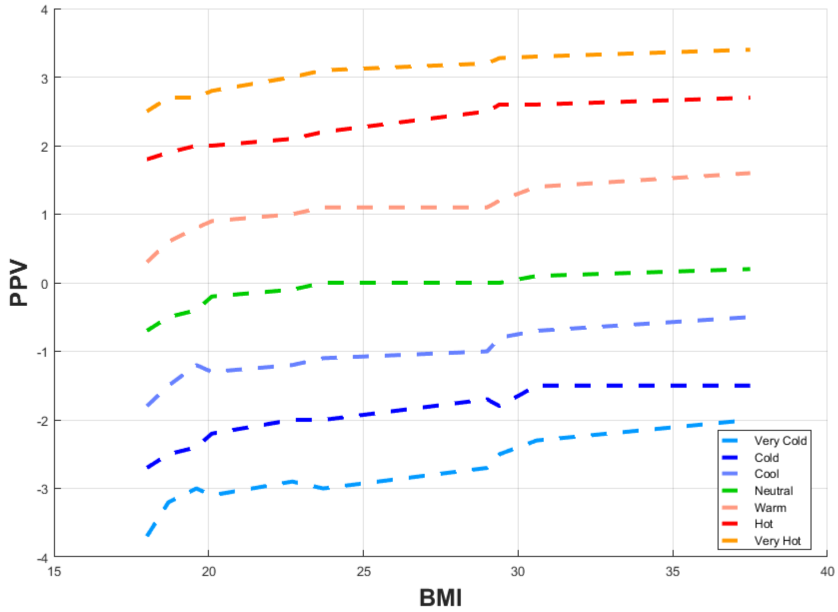


Figure 4.10: The relationship between BMI score and PPV votes.

have assumed an air speed of 0.20 m/s, a constant humidity of 50%, and a metabolic rate of 1.0 (to simulate a seated position).

To link thermal comfort with energy consumption, the ASHRAE handbook [31] was used as a guide, and to calculate the peak cooling loads for one office room. Assume a 16' x 12' office space with a 10' high ceiling. Further, assume the room has curtain walls and U-Factor $= 0.075 Btu/h \cdot ft^2 \cdot^\circ$ C. The U-Factor indicates the heat loss rate of a window assembly such that the lower the U-Factor value, the greater a window's resistance to heat flow and the better its insulating properties. In this case, the U-Factor indicates that the heat coming in from windows is minimal. Regions 1 and 2 in Figure 4.11 (a) show the comfort zones when an occupant has a clothing insulation level of 0.74 clo and 0.5 clo, respectively.

From Figure 4.11 (a) it is evident that if the indoor temperature was set to 21°C and the occupant is wearing 0.5 clo insulation, then they would be outside of their

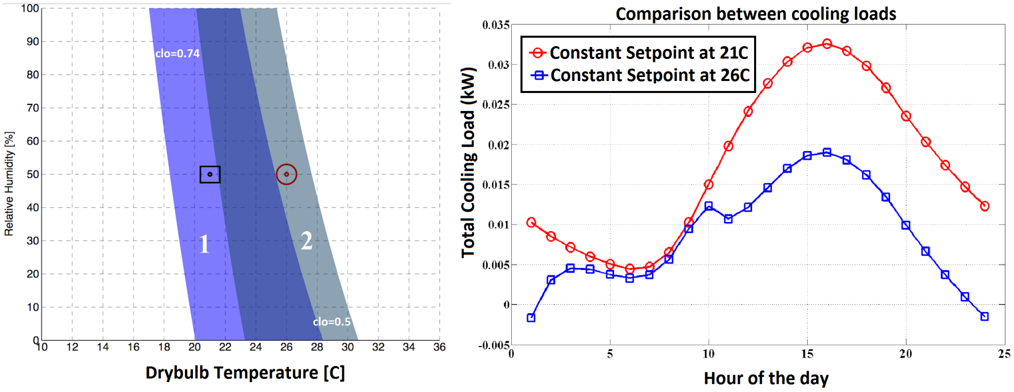


Figure 4.11: (a) Occupant comforts relative to their comfort zone at 21°C (square) and 25°C (circle). (b) Comparison of cooling loads in kW with and without insulation detection

comfort zone (that is, outside of Region 2) and would feel cold. However, when the occupant put on a sweater and increased their clothing insulation to 0.74 clo, they moved inside of their comfort zone (represented by the square in Region 1), and consequently became thermally comfortable. The circles in Figure 4.11 (b) represent the energy usage where one set-point is used throughout the day. Between 12 AM and 6 AM, the energy usage is low as the outdoor temperature is also low, and hence the system requires less work to keep up the 21°C indoor temperature. As the day progresses and the outdoor temperature increases, the system requires more energy to keep up the indoor temperature set-point, and hence energy usage increases. The total energy usage in this case is 10.11 kWh for 24 hours.

Now assume that SiCILIA is deployed in this building, and makes an occupant clothing insulation reading at 11 AM. In this case, SiCILIA has detected that the occupant's clothing insulation is at 0.74 (trousers and sweater). This is an abnormally high level of insulation for an outdoor temperature of 33°C, where the typical insulation level worn by people is approximately 0.5 clo. As such, it is safe to infer that the current indoor temperature is too cold for this occupant to be wearing 0.5 clo (trousers/shorts and a t-shirt). Hence, energy can be saved by increasing the indoor temperature to a value where the occupant is comfortable. From Figure 4.11 (a) it is clear that the new indoor temperature can be set at 26°C, a temperature that would make the occupant comfortable at 0.5 clo. Consequently, the HVAC system will need to utilize less energy for keeping up this set-point. Specifically, it is evident from Figure 4.11 (b) that the system will save energy as the new indoor temperature is closer to the outdoor temperature, and hence requires less energy to keep up. In the case of the new 26°C set-point, the clothing-aware system spent an average of approximately 5.0 kWh/day, resulting in energy savings of approximately 49% over a 24 hour period. This back-of-envelop calculation shows that additional energy savings can be achieved by detecting the deviation of the clothing level from "normal" wear given the outdoor temperature. However, it should be noted that the CBE calculator is based on PMV and does not account for individual preferences.

4.9 Summary

In this chapter, we present the design and implementation of a low-cost smart sensor (SiCILIA) for the inference of human clothing. Specifically, SiCILIA measures environmental parameters such as the ambient temperature, humidity, target skin temperature, and target clothing temperature, and finds the occupant's level of clothing insulation. We have deployed SiCILIA in a temperature-controlled vehicle in order to create a challenging, transient environment in which to test its performance. Results show that SiCILIA is capable of predicting the occupant's clothing insulation level with an RMSE 0.08 with properly tuned parameters.

Chapter 5

CRONOS: A Post-hoc Data Driven Multi-Sensor Synchronization Approach

The increasing popularity of wearable technology and the Internet of Things (IoT) has contributed greatly to the rise of ubiquitous computing in which small, inexpensive, networked processing devices are distributed at all scales throughout everyday life. This has created environments that contain ubiquitous sensor devices with different clock generation methods and communication protocols. Additionally, many applications require the fusion of multiple sensor's data. These sensors may be instrumented on different physical objects or humans, or may be of different vendors or sensing modalities.

Properly aligning the data streams coming from these sensors is crucial to make sense of the data. For example, a person with a heart condition may wear a prescribed heart-rate monitor as well as a personal activity tracker (i.e. a fitbit, Garmin fitness tracker, etc.). The patient is typically asked to journal her activities over 24 hours, which can be a cumbersome task and prone to human error. Instead, it would be more accurate and less intrusive to implement an activity inference algorithm based on the activity tracker reading and use the results to annotate the heart rate data collected during the 24 hours. The physician can then interpret the heart-rate data more accurately. Furthermore, consider a different scenario in which multiple people with a wrist-worn activity tracker interact with a door with a motion sensor. If the scenario involved only one person opening the door, it becomes trivial to construct an activity inference algorithm and find out when the door was opened by this person. However, when multiple people are involved it becomes necessary to synchronize between their sensors and the door to accurately find out who opened the door. Indeed, even though SiCILIA and XTREMIS are designed to function as standalone systems, they too would require synchronization if used in tandem with other sensors or with themselves. For example, if using SiCILIA in a building context along with HVAC sensors, or if using multiple XTREMIS boards at the same time.

Traditional synchronization methods aim to synchronize the clocks on sensors and generally rely on wireless communication protocols (e.g. RBS [16] or GPS) or modifying hardware (e.g. internal real-time clocks (RTC)). Re-engineering entire systems to facilitate synchronization can quickly become costly as the number and types of devices grow. Further, adding more hardware to existing systems (i.e. upgrading with an RTC or a high quality oscillator) is likely to increase cost and power consumption. Indeed, frequent wireless communication may consume too much power, rendering real-time synchronization using network-based methods infeasible for battery-powered devices. Finally, if the data has been already collected without synchronization, network-based methods cannot be applied to it.

It is therefore important to develop a framework that is capable of synchronizing devices post-hoc, without needing to re-engineer their hardware or software. Rather, it exploits the very thing that every sensor collects: *data*. In this chapter, we exploit the independence and omnipresence of motion artifact information in most sensor data. Indeed, cameras and biosensors such as electroencephalographs (EEG), electromyographs (EMG), and electrocardiographs (ECG) can capture motion data as it is prevalent in the $< 20Hz$ range of the waveform and even potentially higher for some motions [21]. Although not strictly granular, there is enough information in the data to infer whether there is movement or not. However, post-hoc synchronization presents multiple challenges:

- How to identify multiple sensors detecting the same movement?
- For non-rigid objects, movements on different parts may be not only be out of sync, but out of phase.
- How to extract motion from these non motion-oriented sensing modalities?

We thus present CRONOS: a post-hoc, opportunistic data-driven framework for sensor data synchronization that takes advantage of common events observed by multiple sensors.

CRONOS identifies and extracts common events that are suitable for synchronization by first extracting motion information from the synchronizing sensors' data streams. The skew and offset in time between the sensor data streams are then estimated and their timings adjusted accordingly. For the multi-sensor case, CRONOS utilizes a graph-based approach using the most up-to-date synchronization data. This optimization leads to further reduction in synchronization errors. To demonstrate its versatility, we apply CRONOS to both pairwise and multi-sensor motion, optical, and key press data synchronization and achieve a drift improvement of approximately 88% to 97% in both pairwise and multi-sensor synchronization.

The remainder of this chapter is organized as follows. Section 5.1 reviews related works on sensor synchronization techniques. Section 5.2 presents a formulation of the synchronization problem CRONOS is designed to resolve. Section 5.3 presents the overall approach, classification, and pairwise synchronization. Section 5.4 discusses the multi-sensor synchronization protocol. Section 5.5 covers the experiments and gives an analysis of the results obtained. Finally, Section 5.6 presents the conclusion and future work.

5.1 Related Work

There have been many approaches to time synchronization in sensor networks [16, 50, 55, 79, 95, 109]. They can be split into two categories: network synchronization and data-driven synchronization.

With network synchronization, sensors are synchronized in real time using wireless communication protocols. For example, Network Time Protocol (NTP) synchronizes all participating devices within a few milliseconds of Coordinated Universal Time (UTC) [61]. Another example is Reference-Broadcast Synchronization (RBS) and Post-Facto Synchronization (PFS), developed by Elson *et al.* [16]. RBS is a form of time synchronization in which receivers use the arrival time of a reference broadcast to compare their clocks. Elson *et al.* extended RBS to multi-hop RBS in which clocks across broadcast domains are unionized without sacrificing the receiver-to-receiver property of RBS [16]. Finally, PFS is an energy-aware time synchronization technique that lets clocks run naturally until an event of interest occurs, at which time the clocks synchronize. Maróti *et al.* proposed the flooding time synchronization protocol (FTSP) for wireless sensor networks. FTSP synchronizes the time of the sending node to multiple receivers by utilizing a radio message time-stamped at both the sender and receiver sides [58]. However, network synchronization methods are not suitable for synchronizing stored data and require common communication protocols on all devices.

Data-driven synchronization, on the other hand, is less studied. Time delay estimation between signals received at two spatially separated sensors was developed by Knapp *et al.*, as one of the first approaches to sensor synchronization. A maximum likelihood (ML) estimator was developed as a pair of receiver pre-filters followed by a cross correlation. However, the proposed estimator only functions on pairwise sensor time approximation and was not developed for multi-sensor synchronization. Lukac *et al.* proposed data driven time synchronization (DDTS) to synchronize seismic activity sensors [56]. By extracting underlying independent and omnipresent phenomena in the data, a model of their travel time between two earthquake sensing stations is developed. The model is then used to predict the travel time of an incorrectly synchronized station and use it to derive a time correction [56]. Compared to Lukac *et al.*'s work – where all sensors must be capable of sensing Microseisms – CRONOS extracts independent motion artifacts from different sensing modalities to synchronize among different sensors. Fridman *et al.* proposed a method for the automated synchronization of vehicle sensors [22]. Specifically, Fridman used cross-correlation of accelerometer, telemetry, audio, and dense optical flow from three video sensors to synchronize all the sensors offline. All sensors were purposefully instrumented on the same vehicle. Although cross-correlation is applied to the data to correct for offsets, this approach does not take into account the skew of the data. CRONOS utilizes linear regression to compensate for skew, making it more applicable for longer periods of time as drift over long periods of time is quasi-linear. Finally, Bennett *et al.* presented two offline data-driven methods for pairwise synchronization [5]. The first approach utilized the notion of templates, and the other utilized entropy [5].

Both approaches estimate the delays of one clock oscillator vs. the other. The authors acknowledge that a limitation of their technique is the possible matching of the wrong data points between the two sensor streams. This is especially evident in highly periodic signals, however they argue that single point errors do not propagate synchronization error [5]. CRONOS, on the other hand, utilizes cross-correlation to eliminate any misclassified data streams prior to performing synchronization. Bennett *et al.* further extended the work and proposed a methodology to synchronize data streams from multiple sensors. Similarly to CRONOS, the approach exploits common events observed by the sensors. Using a graph model to represent the couplings between sensors and the drift in sensor timing, a solution that employs shortest path to minimize the clock drift in the system is proposed [6]. Bennett *et al.*'s algorithms focus primarily on accelerometer data. Further, to set up the observation windows properly and find alignment points between sensors it is necessary to know the parts-per-million (PPM) of the sensor, which is a measure of the sensor's stability over time. PPM is not always available prior to synchronization and thus limit the applicability of their approach.

It can be surmised that a fundamental difference between network and data-driven synchronization is that it is possible to opportunistically synchronize using the latter, but it is not possible with the former. In other words, it is possible to exploit events that are observed by multiple sensors at the same time for synchronization; whereas in network synchronization useful events are pre-defined and generated by the protocols themselves.

5.2 Problem Formulation

CRONOS is designed to synchronize the data from multiple sensors. We assume the sensor data are associated with timestamps according to local clocks, thus we use the terms "local clock timestamp" and "sensor clock" interchangeably. Additionally, we define the speed of sensor i's clock relative to sensor j's as the "relative skew", and the delay between sensor i's clock relative to sensor j's as the "relative offset".

Table 5.1: CRONOS notations

Variable	**Description**
t_i	Local time of sensor i
t	Real time
α_{ij}	Relative skew of sensor i's clock to sensor j
O_{ij}	Relative offset sensor i's clock to sensor j
x	Data stream

Figure 5.2 shows the drift between two sensor clocks over 31 hours. The quasi-linear nature of the curve indicates that it is inaccurate to assume sensor drift is purely linear.

However, from Figure 5.2 and findings by Bennett *et al.* [5, 6] it is reasonable to assume that over a short period of time (that is, minutes or a few hours) the drift

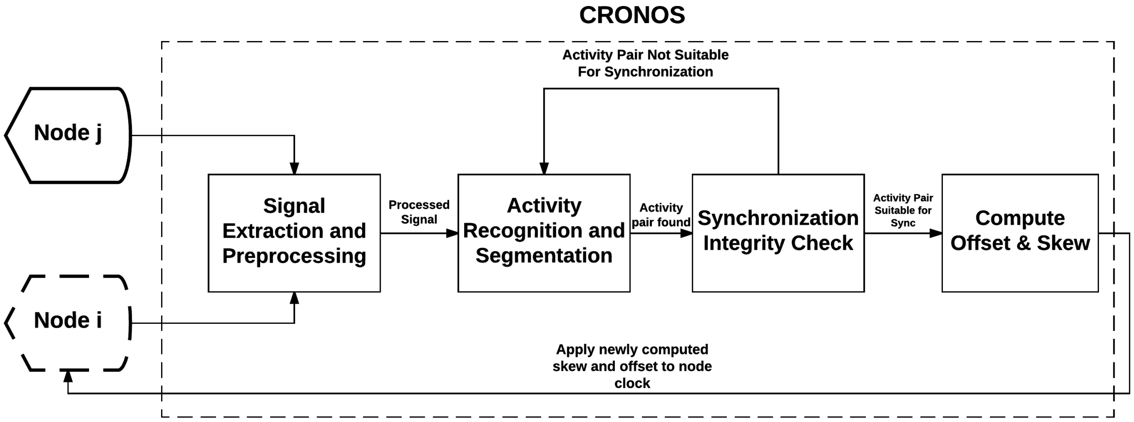


Figure 5.1: The flow of CRONOS to synchronize a pair of nodes,where node j is the reference node and node i is the node in need of synchronization

is indeed linear. If the data streams from these sensors are left unsynchronized, the timing of events observed from both sensors may differ greatly, even if the sensors recorded the events at the same time. We introduce Synchronization Opportunities (SOs) as the common events among sensors during which synchronization is possible. In the heart-rate scenario, SOs can be the increased heart rate on the heart-rate monitor and the jogging activity recorded on the activity tracker, whereas in the door scenario it would be the twisting of the door knob and the user's wrist. In order to identify SOs, CRONOS utilizes activity recognition to find the general event start and end times. We define an event as a singular occurrence detected by the sensor such as taking a step when a person is walking. On the other hand, we define an activity as a *category* consisting of a singular event or repetitive identical events. For example in Figure 5.3, walking is an *activity* – labeled "Activity 1", which consists of 11 steps, regarded as *events*. In this work, we exploit walking, pressing a keyboard key, and handshakes as SOs.

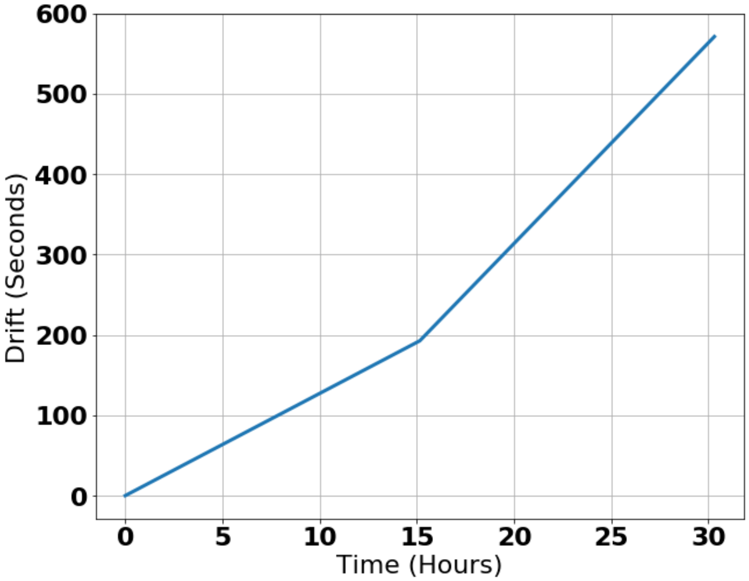


Figure 5.2: Drift reported between two sensor nodes over 31 hours.

To synchronize a pair of sensors, we must find all the common observations between both sensors and mark them all as SOs. Consider two sensors s_i and s_j observing activities together. To synchronize between them, all SOs must be identified. Once the timings of all SOs are determined among the data streams from both sensors, one can apply linear regression to determine the relative skew and drift. For multi-sensor synchronization, CRONOS handles the quasi-linear nature of sensor drifts by utilizing a graph-based approach to synchronize from one sensor to the next most recently synchronized sensor.

5.3 Pairwise Synchronization

The flow of CRONOS for pairwise synchronization is summarized in Figure 5.1. In the case of varying clock skews the synchronization algorithm is applied over a moving window of the data streams. The incoming data streams are pre-processed to extract

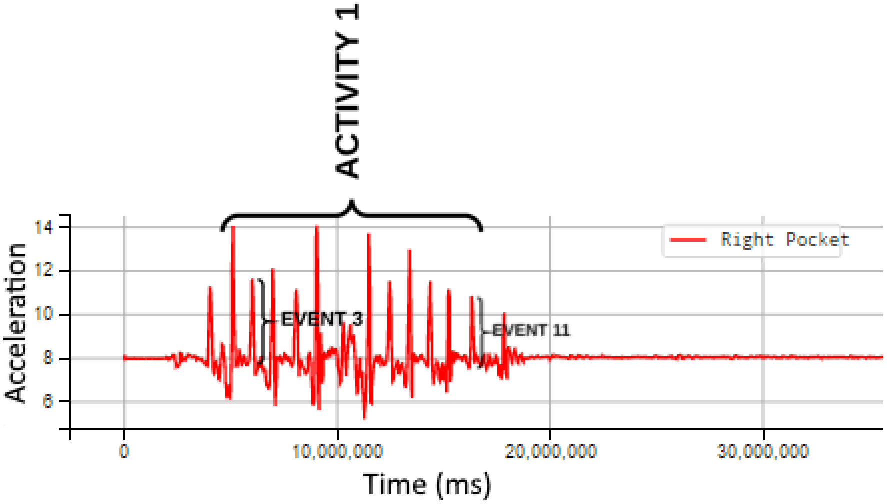


Figure 5.3: Walking activity consists of taking multiple steps, each of which is regarded as an event.

useful signals and then fed into a classifier for event detection. On nodes i and j, each activity and its events found by the classifier are extracted and prepared for cross correlation. Cross-correlation is first applied to the preprocessed waveforms corresponding to common activities found on nodes i and j to find the overall offset O_{ij} (or O_{ji}). Cross-correlation is then applied to the individual events within the activities to build a linear model to estimate the skew α_{ij} (or α_{ji}).

5.3.1 Extraction of Movement Data

Movement data can be captured by many sensing modalities from cameras to EMG in addition to accelerometers or vibration sensors. Indeed, the conventional wisdom is to treat motion artifacts as noise to be removed, however CRONOS takes advantage of these artifacts as they are independent, omnipresent components in the data from wearable or IoT devices that can be used for synchronization. In EMG for example, motion information is present in the signal at below $10 - 20Hz$. Figure 5.4 shows an example of extracting motion information from one EMG channel using an EMG device mounted on the user's forearm as they typed a paragraph on a keyboard.

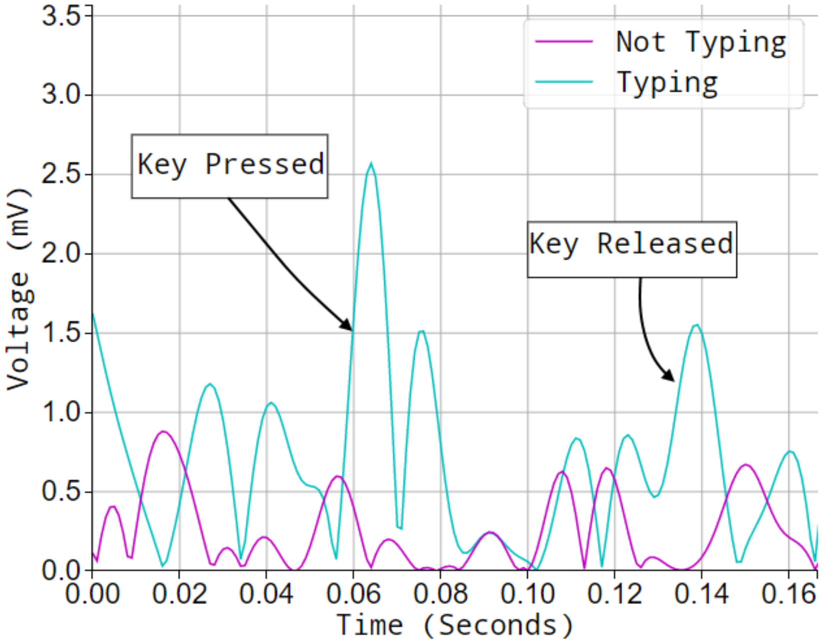


Figure 5.4: Comparing the presence of motion artifacts in one EMG channel on the user's forearm during a typing activity v.s. not typing.

Cameras also capture motion indirectly. Optical flow is typically used to derive the displacement of objects in video data. Figure 5.5 shows a comparison between accelerometer data and the optical flow data from a helmet-mounted accelerometer and camera worn by a user as they walked towards an object. Note that the vibrational data from walking is evident in both the accelerometer magnitude and the optical flow along the Y axis, where each peak above 3 pixels of deviation is a step taken.

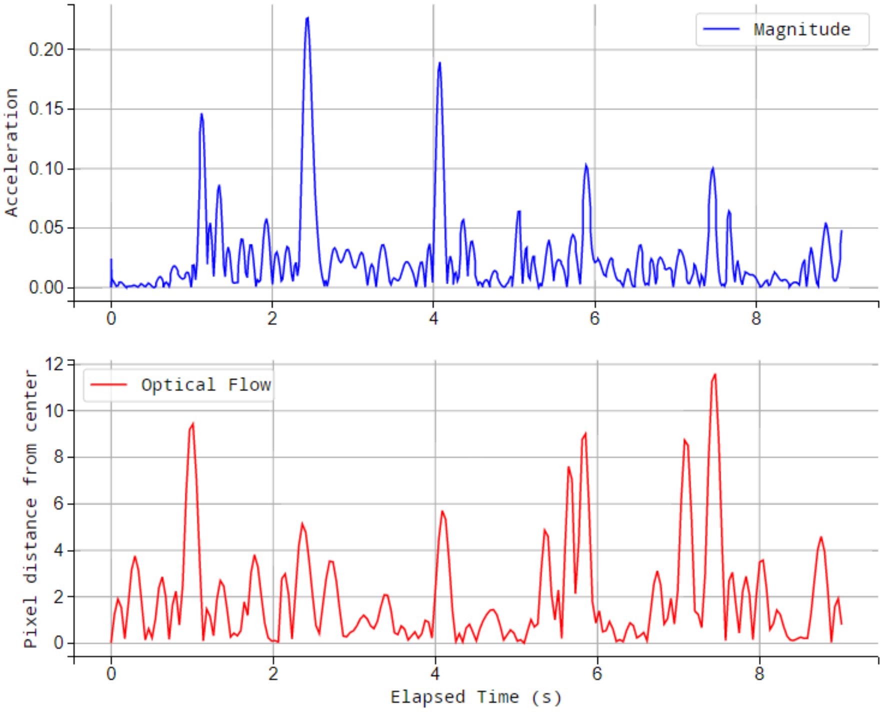


Figure 5.5: Walking data collected by a participant wearing an accelerometer and a camera-mounted helmet.

Compensation for Phase Differences

Sometimes sensors may be placed in symmetrical locations. If such phase shifted activities are left unchecked and the associated waveforms are used directly in synchronization, the result will be off by at least half a gait cycle, or 500ms in the example in Figure 5.6 (top). This is clearly unacceptable. However, we observe that not all frequency components of the accelerometer data are subjective to the same degree of phase shift between the two sensors. In particular, when a person's foot hits the ground, vibrations that permeate the entire body are picked up by the accelerometers. The characteristic frequency band of the vibrations generated by the force normal to the ground/floor has been shown to be approximately $20 - 90Hz$ [15]. CRONOS extracts these vibrations by applying a high-pass filter at $> 20Hz$ to remove the gait cycle information, treating it as noise instead.

5.3.2 Identification of SOs

As we previously defined, SOs are the possible observations among sensors during which synchronization is possible. Bennett *et al.* referred to this phenomenon as alignment points and used template-based and entropy-based methods to find them [5, 6, 7]. The limitation of this approach, however, is that it requires prior knowledge of the sensor's PPM – a measure of a sensor's stability over time. This information may not always be available or worse – if the PPM specified is incorrect then the synchronization effectiveness will suffer. Additionally, template- and entropy- based methods have a higher chance of false positives or false negatives than a discriminant classifier. Therefore, CRONOS utilizes a support vector machine (SVM) to identify the activities being performed. Additionally, within the context of this framework

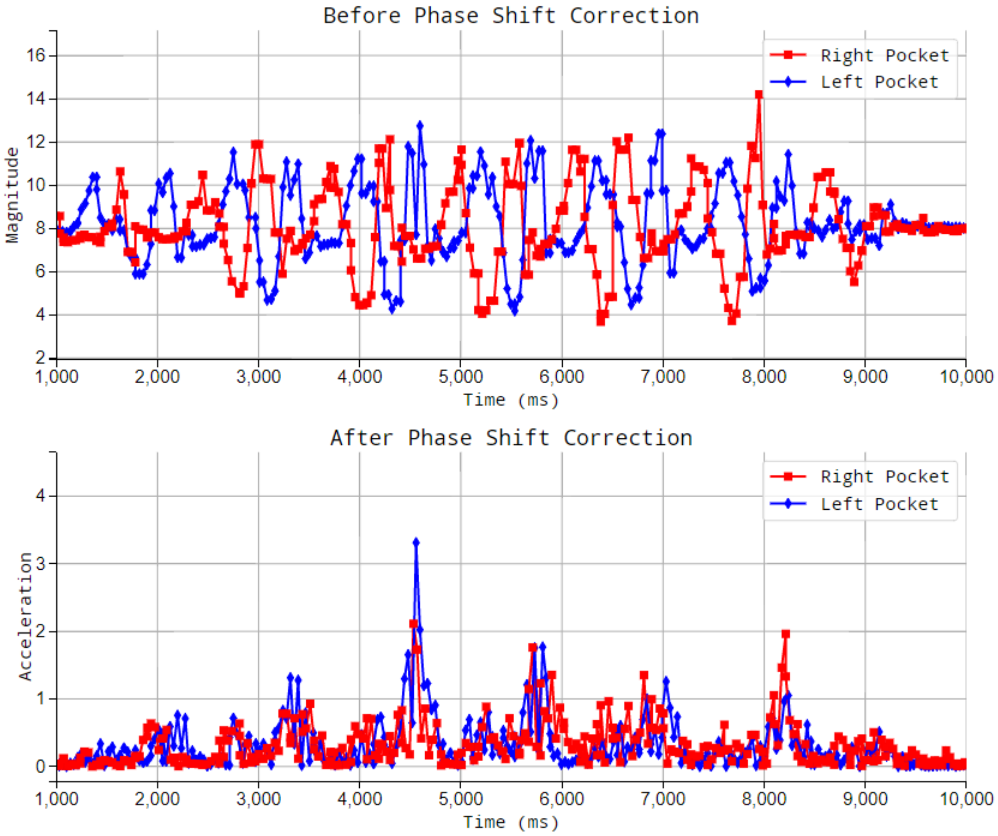


Figure 5.6: Walking data from two accelerometers (in right and left pockets) before and after applying a high pass filter to obtain the inherent vibrations that permeat the human body.

a classifier can be swapped in and out for different activities to be identified. For example, a trained classifier may be used to find motion-based activities but it can be swapped out for a different classifier to identify muscle-based activities.

Activity Recognition

Since the classification must be as accurate as possible to avoid misclassifying events, it is important to build a robust feature set. In this work, we adopt some of Hudgin's feature set described in Chapter 3 due to the simplicity of their computations and their ability to describe the signal in the time domain well. Specifically, the mean absolute value, distance mean absolute value, sign slope changes, and the waveform length were utilized. Further, due to its high dimensionality, memory efficiency, and versatility, a SVM was utilized to classify the incoming data streams.

5.3.3 Integrity Check

Prior to synchronizing, the detected pairs of activities (or events) must first be transformed to remove all potential phase differences, then verified as suitable for synchronization. However accurate the SVM classifier is, it can still be erroneous. Misclassification of the activities may lead to large synchronization errors. Generally, there is a trade-off between false positive and false negative rates. In CRONOS, it is desirable to have low false positives rates while moderate false negative rates only imply synchronization opportunities but would not negatively impact synchronization accuracy. We define synchronization integrity (SI) as the likelihood to assess whether activities or events identified on the data from a pair of sensors are indeed the same. We identify three cases in which the SI may be compromised:

1. *False Negative Classification*: The classifier has missed an activity or event that should have been caught on one of the sensors.
2. *False Positive Classification*: The classifier has determined that an activity or event has occurred on one of the sensors when in reality it has not.
3. *Composite Activity*: Activities may become intermixed with other activities happening simultaneously (e.g. texting and driving). In this case, activities across two sensors may be the same but will contain different events.

Besides confidence measures from the classifier (e.g. SVM can compute the distance from a feature vector to a decision plane), we can also evaluate SI at the signal level. Intuitively, the closer the two waveforms associated with the identified common activities are, the more likely these activities captured by different sensors are indeed identical. We propose two approaches: normalized cross-correlation (NCC) and dynamic time warping (DTW), for measuring signal similarity.

NCC (described in further detail in Section 5.3.4) is essentially the dot product between the two normalized vectors. The output can be defined such that $1 > NCC > -1$, where the closer to 1 the vectors are, the more similar they are. If NCC is closer to 0 or -1, then the vectors are orthogonal to each other or are pointing in opposite directions, respectively.

On the other hand, DTW is an algorithm for measuring the similarity between two time-series sequences regardless of their speed. More formally, DTW aligns two time-series data streams x_i of length n and x_j of length m from nodes i and j where:

$$\begin{aligned} x_i &= a_1, a_2, ..., a_N, \\ x_j &= b_1, b_2, ..., b_M. \end{aligned} \tag{5.1}$$

An $N \times M$ matrix is then constructed where the (n^{th}, m^{th}) element of the matrix corresponds to the squared distance $d(a_n, b_m) = (a_n - a_m)^2$ which is the alignment between points in x_i and x_j [75]. A path through the matrix that minimizes the total cumulative distance between these two sequences is retrieved to find the best match between x_i and x_j. Specifically, the optimal path is the path that minimizes the warping cost:

$$DTW(x_i, x_j) = min\sqrt{\sum_{k=1}^{K} w_k}, \tag{5.2}$$

where w_k is the matrix element $(n, m)_k$ that also belongs to the k^{th} element of a contiguous set of matrix elements that represent a mapping between x_i and x_j, referred as W [75]. Dynamic programming is then used to find the warping path by evaluating the following recurrence:

$$\begin{aligned} \zeta(n, m) = d(a_n, b_m) + \\ min\{\zeta(n-1, m-1), \zeta(n-1, m), \zeta(n, m-1)\}, \end{aligned} \tag{5.3}$$

where $d(a_n, b_m)$ is the distance found in the current matrix cell, and $\zeta(n, m)$ is the cumulative distance of $d(a_n, b_m)$ and the minimum cumulative distances from the three adjacent cells.

Comparing DTW and NCC

Figure 5.7 highlights an experiment in which a user walks three times then removes their phone from their pocket and makes a call while still walking (activity 3), the duration of the call is plotted in red in the figure. In this case the classifier may find all matching activities on both sensors. However, activity 3 on the right leg is not similar enough to activity 3 on the left leg and will result in inaccurate event detection when calculating the skew. Table 5.2 gives the output of DTW and NCC in comparing the integrity of activities 1, 2, and 3 in Figure 5.7. Both approaches respond similarly in detecting structurally sound pairs of activities, however in some cases the versatility of DTW may allow users to avoid needing to up-sample the slower data stream. Moreover, it is possible to create a finer resolution by using windowing to compare the signals instead of the entire activity. Indeed, this would allow CRONOS to discard data at the event level rather than the activity level. Finally, this approach is applicable to the other two cases in which SI may be compromised, as false positive and false negative classifications will consequently create dissimilar windows of the signal for synchronization.

5.3.4 Estimation of Skew and Offset

Now that the common activities are identified, we describe the procedure to estimate clock skew α_{ij} and offset O_{ij} using sensor signals. The offset between two nodes i and j is detected using normalized cross-correlation, which provides a quantitative assessment of the similarity of two functions at all possible time shifts. Formally, the cross-correlation between data streams x_i and x_j is defined as:

$$\mathrm{NCC}(k) = \frac{\sum_{n=0}^{N-1} x_i(n) x_j(n+k)}{\sqrt{\sum_{n=0}^{N-1} x_i^2(n) \cdot \sum_{n=0}^{N-1} x_j^2(n+k)}} \tag{5.4}$$

where N is the number of samples in x_i and x_j, $x_i(n)$ and $x_j(n)$ are the samples from data streams x_i and x_j at sample n, respectively, and finally k is the shift between x_i

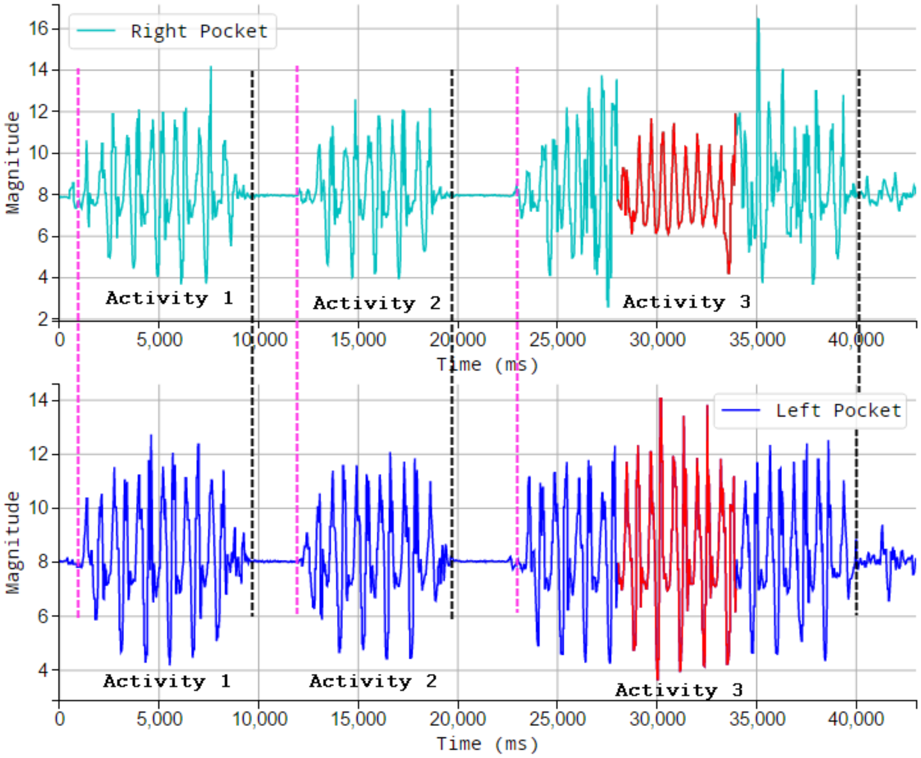


Figure 5.7: Walking data obtained from accelerometers on the right and left legs with random stops. Activity 3 (walking) is composited in that the accelerometer was removed from the pocket and placed near the ear during a phone call.

	Walking	Walking fast	Walking (on call)
NCC	0.74	0.85	0.54
DTW	0.12	0.11	0.34

Table 5.2: Activity integrity output comparison between dynamic time warping and normalized cross correlation

and x_j.

Unlike offset estimation which considers an entire activity when applying cross correlation, skew estimation is more fine-grained in that it considers individual events. Specifically, the offset is calculated at each event detected and a linear regression model is constructed to estimate the skew of sensor i to sensor j. Since the linear model takes the form:

$$t_j = \alpha_{ij} t_i + O_{ij}, i = 1, ..., n, \tag{5.5}$$

where t_j is the time at sensor j, α_{ij} is the estimated skew between node i and node j, t_i is the current time at node i, and O_{ij} is the offset between node i and j.

5.4 Multi-Sensor Synchronization

As previously discussed, the drift experienced by two sensors over long periods of time is not strictly linear. Figure 5.2 shows the drift between two sensors over approximately 31 hours. The variations in the slope over time indicate that the skew is not consistent over long periods of time. Additionally, some sensors do not have direct interactions to observe common events for pairwise synchronization, or the synchronization between them may be erroneous or stale. A more sophisticated, multi-sensor synchronization approach is required in this case. Let α_{ij} and O_{ij} be the real offset and skew between nodes i and j, respectively. The problem is this: We wish to obtain estimates α_{ij} and O_{ij} of the skews and offsets, respectively. Note that the true α_{ij} and O_{ij} satisfy the global constraints:

$$\alpha_{i_1 i_2} \cdot \alpha_{i_2 i_3} \cdot ... \cdot \alpha_{i_m i_1} = 1, \tag{5.6}$$

$$O_{i_1 i_2} + O_{i_2 i_3} + ... + O_{i_m i_1} = 0, \tag{5.7}$$

where $i = 1, ..., m$ is the index of the sensor in a loop made of m sensors in the network. These constraints ensure that if the synchronization goes back to the original node, then the skew and offset must be equal to 1 and 0, respectively. However, the estimates α_{ij} and O_{ij} arrived through only pairwise synchronization described in Section 5.3.4 do not need to satisfy these constraints. Further constraints include:

1. The skew and offset at time t at each sensor node is dependent on its neighboring nodes. In the graph, nodes become neighbors attached by an edge if there was a direct pairwise synchronization between them. Hence, a node's relative skew and offset is directly influenced by its neighbors' clocks.
2. The network is connected.

We define a graph model of the system

$$G = (V, E) \tag{5.8}$$

where the vertices V represent the sensor nodes and edges E represent a direct pairwise synchronization between the nodes. Further, let N_i denote the neighbors of node i and $|N_i|$ the total number of the neighbors. Assume that we have estimates of $\hat{\alpha}_{ij}$ and $\hat{O}_{ij}$ for $j \in N_i, \forall i$. The estimates need not exactly be equal to the true values.

We take advantage of the edge weights in the graph to represent the validity of synchronization. The higher the edge weight, the more up to date the synchronization is and therefore more likely that edge will be to be taken in the path from node i to node j. Consequently, weighted least squares is used to estimate the skew and offset and define the weight for an edge between node i and node j as:

$$w_{ij} = \frac{\delta_{ij}}{\sigma_{ij}} \tag{5.9}$$

where σ_{ij} is the standard deviation of the skew estimates so far between node i and j, and

$$\delta_{ij} = exp^{\frac{t-s_{ij}}{c}} \tag{5.10}$$

in which t is the current time, s_{ij} is the time of the last synchronization between nodes i and j, and c is a unit converter that can be used to convert time to seconds, minutes, days, etc.

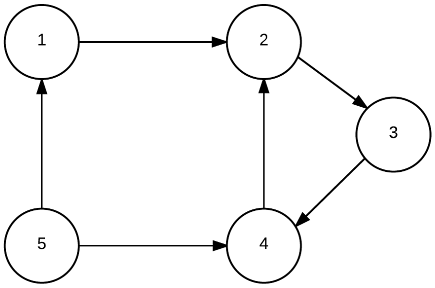


Figure 5.8: Example graph in which each vertex corresponds to a sensor and each edge represents a synchronization between them

As an example, there are six nodes in the graph in Figure 5.8: $1, 2, 3, 4, 5, 6$. We define the oriented incidence matrix A as with arbitrarily assigned edge directions:

	(1,2)	**(2,3)**	**(3,4)**	**(4, 2)**	**(5, 1)**	**(5, 4)**
1	+1	0	0	0	-1	0
2	-1	+1	0	-1	0	0
3	0	-1	+1	0	0	0
4	0	0	-1	+1	0	-1
5	0	0	0	0	+1	+1

where in the row corresponding to node i, we have an entry $+1$ for all edges of the form $(i,)$, an entry -1 for all edges of the form $(, i)$, and 0 otherwise. We denote v_i and u_i as the estimate that we will make of the offset and skew with respect to a reference node, respectively. Written as a vector of edge offsets, $A^T v$ and $A^T u$ convert the node offsets into edge offsets. The problem formulation to find skew and offset is:

$$\begin{aligned} &\underset{v}{\text{Min}}\ ||w^T \cdot (A^T v - \alpha)||^2, \\ &\underset{u}{\text{Min}}\ ||w^T \cdot (A^T u - O)||^2, \end{aligned} \tag{5.11}$$

where w is a $|E|{\times}1$ vector representing the weights of the edges, and α and O are $|E|{\times}1$ vectors representing the skews and offsets of the nodes in the graph respectively. Since $w^T \cdot (A^T v - O)$ and $w^T \cdot (A^T u - \alpha)$ are both in the null space of A, the solution to the optimization problems is:

$$AA^T v = AO, \tag{5.12}$$

and

$$AA^T u = A\alpha. \tag{5.13}$$

Initial O and α for each node are assigned arbitrarily. Equations (5.12) and (5.13) can be solved by setting an arbitrary $v_i = 0$ and $u_i = 0$, removing one row from AA^T, and maintaining a $|V|{-}1$ rank. The complexity of solving this equation is $\mathcal{O}(n^3)$ due to the Gassian Jordan matrix multiplication method requiring 3 iterations. There are faster methods that can be utilized such as the Coppersmith–Winograd ($\mathcal{O}(n^{2.376})$) or the Stassen ($\mathcal{O}(n^{2.807})$) algorithms, but the advantage in performance will only be relevant with very large matrices – in this scenario, when the number of sensors and synchronizations between them creates matrices so large that they cannot be processed on conventional hardware. Additionally, since the algorithm is applied offline the time

5.5 Experiments and Results

Due to its clock flexibility and ease of access, a Texas Instruments MSP430 microcontroller was used as each sensor node's core microprocessor during evaluation. For motion sensing, an Invensense MPU9150 measurement unit (IMU) collecting acceleration data at 100Hz was utilized. Each sensor timestamps the data using the on-board crystal oscillator ($\pm 20 ppm$, i.e. 1.01 second drift after 14 hours) as well as the DCO ($\pm 5000 ppm$, 4 hour drift after 14 hours). The DCO has low clock accuracy but is often used in tandem with the crystal oscillator. It is used as a standalone oscillator during experimentation as the drift will occur quicker than using two crystal oscillators. The crystal oscillator is used as the ground truth as the accuracy of the crystal will be high. Each synchronization is applied to sensor data using the crystal and DCO timestamps to obtain both ground truth and drifted data, respectively. We present the results for pairwise synchronization and then present the results for multi-sensor synchronization. The absolute synchronization error is used to evaluate synchronization accuracy in both the pairwise and the multi-sensor case, defined as:

$$\mathrm{ASE}(\gamma_k, \hat{\gamma}_l) = \frac{1}{N} \sum_{i=0}^{N-1} \left| \gamma_{k_i} - \hat{\gamma}_{l_i} \right| , \tag{5.14}$$

where N is the number of samples, γ_{k_i} is the real time at sample i in node k

and $\hat{\gamma}_{l_i}$ is the synchronized time at sample i in node l. Since drift is consistently increasing in a quasi-linear fashion like in Figure 5.2, the absolute synchronization error is obtained at every synchronization action and averaged with other obtained errors to obtain the mean absolute synchronization error.

5.5.1 Data Collection

We consider several scenarios where interactions among the subjects create synchronization opportunities for sensor data.

- Two subjects wore an accelerometer on their wrist and carried on a conversation while emphasizing and gesturing with their hands as they spoke. Subjects would also pick up an item at random times during the experiment and then place it back down to emulate a real conversation and to generate possible false positives. During the conversation, the subjects were asked to shake hands at random to collect handshaking data – the common activities used for synchronization.
- A subject types on a keyboard with an accelerometer mounted on their wrist. During the experiment, the subject typed a paragraph and used their thumb every time they wished to press the space bar. The key press and release were timestamped on the keyboard and accelerometer and then sent to the computer to be recorded. Pressing the space bar with the right thumb was chosen as the synchronizing event because the shortness of the thumb forces the human hand to move more than the rest of the fingers would.
- Subjects wore a helmet fitted with an Arduino Uno-powered camera and an accelerometer was used to collect data when they walks. The camera collects

data at a rate of 10 frames per second (FPS), which is enough to capture walking data as it is typically 1.8-3 Hz [45, 101]. During data collection, the user walks towards an object mounted on a wall. The camera captures the movement of the observer relative to the object, namely the high frequency "bumps" that occur each time a user's foot hits the ground due to the vibrations in the body.

5.5.2 Classification Accuracy

Data from 5 volunteers was collected for conversing, typing, and walking. Each classifier was evaluated using 10-fold cross validation. Table 5.3 highlights the activities utilized for each category of classification and the accuracy, precision, and recall achieved. The column labeled "Target Event" describes the event that is searched for when performing synchronization, whereas the "Interfering events" column describes activities that are added to the training data in order to create an activity recognition classifier that is more robust and discriminative.

Table 5.3: Synchronization activities and the classification accuracy, precision, and recall for each one.

Activity	**Target Event**	**Interfering Events**	**Accuracy**	**Precision**	**Recall**
Conversing	Handshakes	Converse with hands. Pick up item	96.22%	89.05%	87.32%
Typing	Press using thumb	Press using other four fingers	95.12%	95.65%	92.86%
Walking	Steps taken	Turning around, misstep	97.54%	92.78%	89.10%

5.5.3 Pairwise Synchronization

Synchronization between two sensors should be possible regardless of the sensor's modality as long as both sensors can observe the effects of the same event. Each experiment in Section 5.5.1 was performed with 5 volunteers. Figure 5.9 shows the effects of applying CRONOS on pairwise synchronization once between two sensors. Since the synchronization period is small, the drift is linear. If the synchronization period is long then the drift would take on a more quasilinear form, and the adjusted drift would take on a sawtooth-like shape as CRONOS continuously synchronizes. Table 5.4 presents the drift improvement and the mean absolute error for each experiment in comparison with the work by Bennett et al. [5, 6]. In shaking hands, both users are wearing an accelerometer sensor and hence both systems perform well. Indeed, although these performance results were obtained after extracting the motion information from the data, it is likely not a fair comparison between the two systems since Bennett et al.'s implementation suffered as it does not account for different sensing modalities (like the the cases of typing on a keyboard and walking with a helmet-mounted accelerometer and a camera).

Table 5.4: Pairwise results comparison in accuracy and mean absolute error (MAE) between CRONOS and the work by Bennett et. al [5, 6, 7].

		CRONOS		Bennett *et al.* (2017)	
		Drift Reduction (%)	MAE(ms)	Drift Reduction (%)	MAE(ms)
Activities	Shaking Hands	88.01%	5.41	84%	6.33
	Typing on keyboard	91.64%	6.28	82.38%	8.52
	Walking with Helmet	87.12%	6.62	78.35%	11.37

5.5.4 Multi-Sensor Synchronization

To verify the performance of CRONOS for multi-sensor synchronization, a simulation and a small-scale sensor experiment were performed. Besides verifying the performance of CRONOS, the goal of the experiments is two fold: to verify the importance of the synchronization fidelity described in Section 5.4, and to compare its performance against other methodologies such as weighted and unweighted shortest path

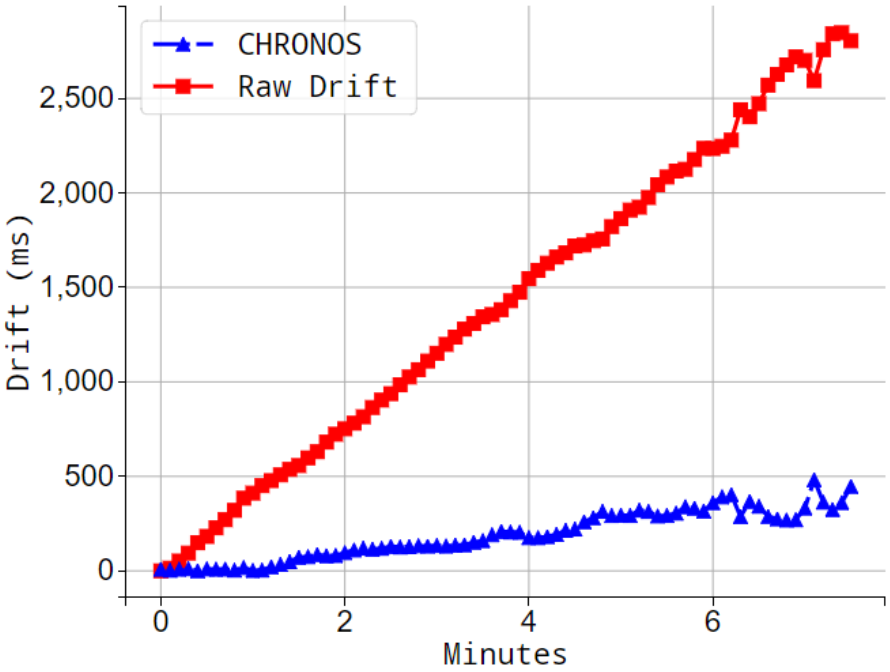


Figure 5.9: Comparison between the raw sensor drift and the sensor drift when applying CRONOS one time.

and opportunistic pairwise synchronization.

Shortest path has been utilized in the past to resolve sensor synchronization across multiple nodes [6]. We compare its performance against CRONOS in both the weighted and unweighted cases to highlight the changes that CRONOS' weighing scheme makes. Further, it is possible to simply wait and perform pairwise synchronization on each pair of sensors. We utilize the following approaches as benchmarks to compare to the performance of CRONOS:

1. **Weighted CRONOS (WC)**: Applying CRONOS with the edge weighing scheme proposed in this work.
2. **Bennett's Approach (BA)**: Applying Bennett *et al.*'s approach [7].

3. **Unweighted CRONOS (UC)**: Applying CRONOS without using edge weights
4. **Weighted Shortest Path (WSP)**: Applying Shortest path between nodes using the same weighing scheme used by CRONOS.
5. **Unweighted Shortest Path (USP)**: Applying Shortest path without using edge weights.
6. **Opportunistic Pairwise (OP)**: Ordinary pairwise synchronization using the first path found from node i to node j.

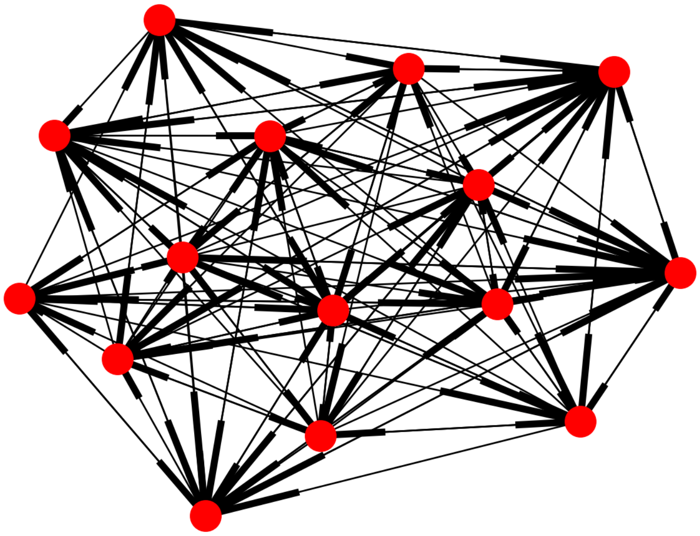


Figure 5.10: Multi-sensor Synchronization Simulation: A randomized 15-node graph generated by the simulation.

Experimental Results

To test the capabilities of CRONOS in a more realistic environment, an experiment was set up to synchronize data between three users interacting with sensors for 30 minutes, using handshakes, walking, and keyboard presses for synchronization. It proceeded as follows:

1. **Users 1 and 2**: wearing accelerometer on wrist.
2. **User 3**: wearing a data collection helmet with an accelerometer and a camera mounted on it, and a wrist-worn accelerometer.

Additionally, a keyboard as described in Section 5.5.1 is set up to collect typing data. The users performed the following actions at random times during the experiment:

1. **Scenario 1**: User 1 shakes hand of user 2.
2. **Scenario 2**: User 3 walks towards users 1 or 2 and shakes their hand.
3. **Scenario 3**: Any user types on the keyboard then walks away.

For scenario 1, there are two sensor data streams: user 1 and user 2's wrist-worn sensors. Scenario 3 has five synchronizing data streams: users 1,2, and 3's wrist-worn sensors, and user 3's helmet-mounted sensors. Finally, scenario 3 has six data streams: all user's wrist-worn sensors, the helmet-mounted sensors, and the keyboard.

Figure 5.11 shows the graph generated by CRONOS when synchronizing all sensors worn by the users and the keyboard. Sensor 1 in this case first acts as a reference sensor, each edge is weighed according to 5.9. Sensors 1 to 4 are the accelerometers worn by all 3 users (where user 3 is wearing one on his wrist and one on his helmet), sensor 5 is the keyboard, and sensor 6 is the camera on user 3's helmet.

Figure 5.12 shows the mean absolute synchronization error in milliseconds, predictably with WC having the lowest error and OP having the highest. BA performs well but still has a higher mean absolute error due to the different sensing modalities

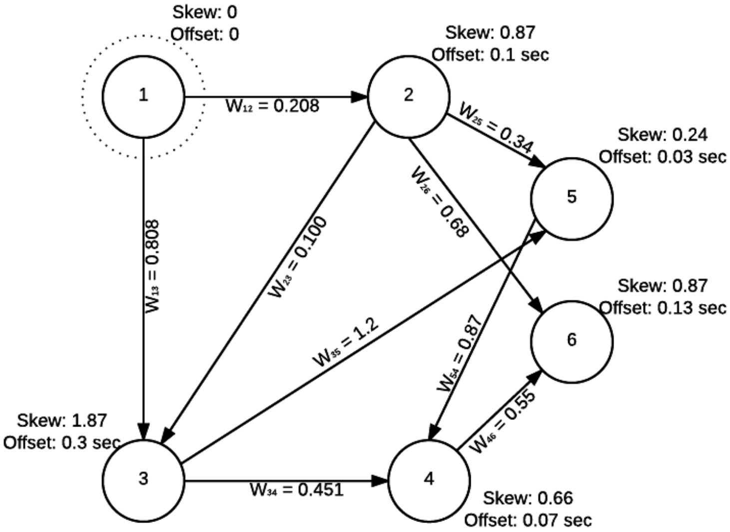


Figure 5.11: Multi-sensor Synchronization Experiment: The graph generated by CRONOS synchronizing the sensors during the experiment

in the system, there also appear to be problems with phase shift handling as the approach performed worse on walking data.

Simulation Results

A 15-node network was constructed as a simulation of optical flow, acceleration, gyroscope, and EMG sensors in which each clock's skew (with respect to the simulation clock) was adjusted slightly over time by a random value. This simulates the behavior of the sensor clock drift similarly to Figure 5.2. As simulation ran, edges were created between the nodes at arbitrary times and assigned a weight using 5.9. The simulation ran for 32 hours and synchronization was run every four hours. Figure 5.10 shows a 15-node graph generated by the simulation, with each node having its own offset and skew. Each edge in the graph is weighed using (5.9), where the thick part of the

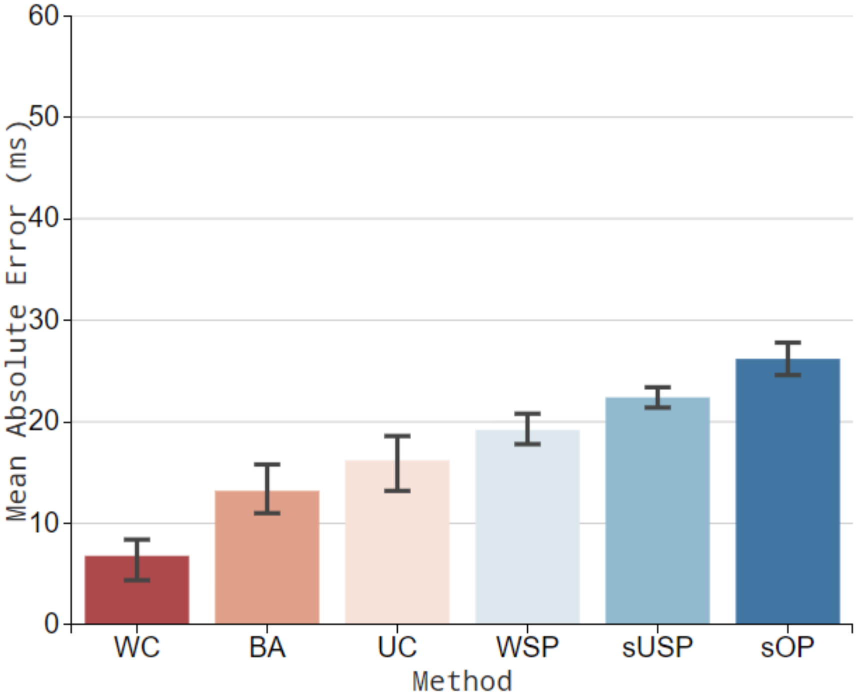


Figure 5.12: Multi-sensor Synchronization Experiment: The drift reduction achieved by CRONOS vs. other approaches.

edge indicates the edge direction.

Figure 5.13 shows the simulation performance under different densities of the graph. The graph density is defined as:

$$D = \frac{|E|}{|V|\cdot(|V|-1)} \tag{5.15}$$

In this experiment, the number of edges determines the graph density and the number of vertices remains the same. As the number of edges decreases, the mean absolute synchronization error increases overall. This is expected as a lesser number of edges means less synchronizations and less common events observed by different sensors. However, given the full 143 edges, CRONOS performs better than the real-world experiment. Additionally, the trend in the performance of the different methods remains the same, indicating that not only does WC perform best, but that it still outperforms traditional methods and Bennett *et al.*'s method. However, the simulation and experiment datasets contain data from sensing modalities other than accelerometers and complex activities that may have affected the performance of Bennett *et al.*'s approach.

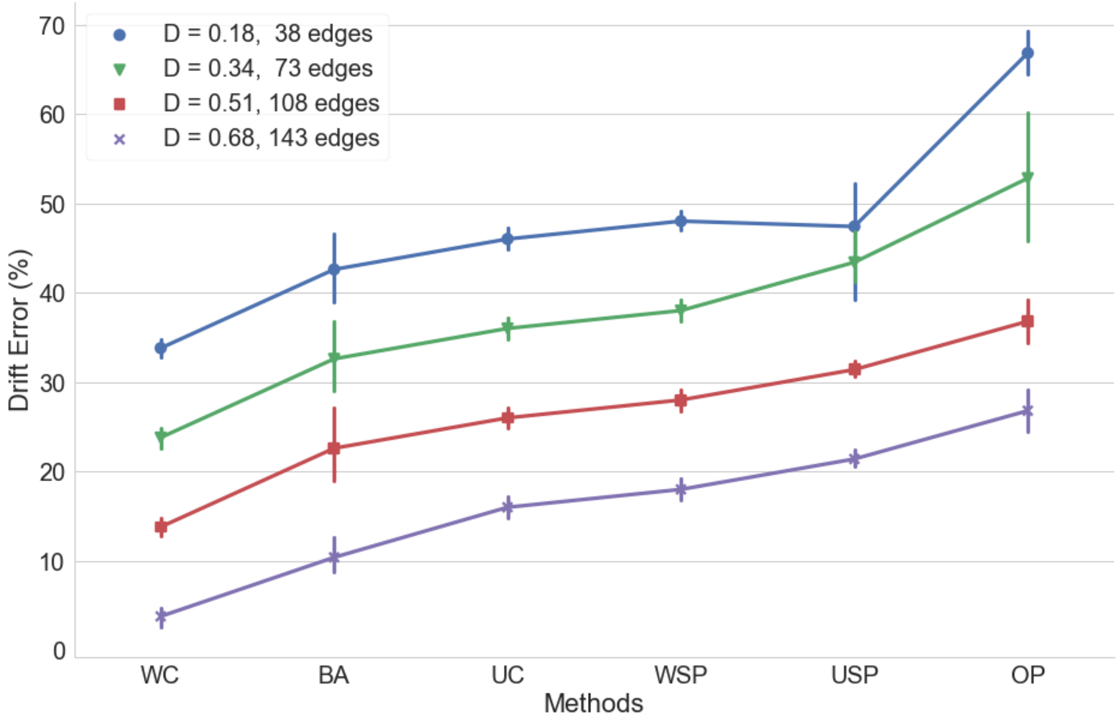


Figure 5.13: Multi-sensor Synchronization Simulation: The drift reduction achieved by CRONOS vs. other approaches.

5.6 Summary

Traditional synchronization approaches require more power and cannot be applied opportunistically or after data collection, hence there is a need for a post-hoc framework to synchronize data offline. We presented CRONOS: a post-hoc, opportunistic data-driven sensor synchronization framework that takes advantage of events observed by two or more sensors, CRONOS estimates the skew and offset between the sensor clocks and adjusts their timing accordingly. Experimental and simulated results on pairwise and multi-sensor synchronization show a drift improvement as high as 98% for total drift, a mean absolute error of 3ms in simulations and 6ms in real-world experiments.

Chapter 6

Conclusion and Future Work

Worker comfort analysis performed by health and wellness professionals is rather cumbersome (e.g. type on a keyboard repeatedly for ergonomic analysis, sit while body measurements are being taken for posture analysis, etc.). The objective of this work therefore is to develop systems that are capable of analyzing body emission information for potential use by industry professionals to maximize and analyze worker comfort. However, systems that can be used for this type of analysis are rather expensive as they are highly precise and are built with many patented components.

In this work, we presented low-cost and high performing tools for human-centric sensing to collect physiological data and make predictions about user physical states or actions. The proposed systems fall into two categories: contact and contactless systems. Additionally, we proposed a framework to synchronize the data streams from multiple sensors post-hoc. The high performance of these systems suggests that it is possible to attain a similar accuracy to the research-grade counter-parts by properly utilizing machine learning algorithms and inferences. The works in this thesis have been made publicly available on Github [84, 86, 93] for replication or extension by others in the future.

6.1 Contributions

The contribution of this dissertation is that we provide a structured and comprehensive set of end-to-end solutions for human-centric sensing that cover subjects ranging from hardware design, to software architecture, to empirical evaluations. Specifically, we have:

- Designed and implemented a contact-based solution to heart, muscular, and brain activity measurements (Chapter 3).
- Identified the effects of sampling rate, electrode placement, and movement speed on fine-grained gesture recognition (Chapter 3).
- Identified the problem of ballistic gestures and proposed a Gaussian Mixture Hidden Markov Model to take advantage of the finger movement patterns in training data (Chapter 3).
- Designed a contactless desktop solution to track users and measure their clothing insulation levels (Chapter 4).
- Designed an algorithm for the detection of clothing insulation given several parameters such as skin, ambient, and clothing temperatures, clothing emissivity, and others (Chapter 4).
- Designed a software framework to synchronize multiple sensors using only their data in a post-hoc manner (Chapter 5).

In designing end-to-end platforms for human-centric sensing, the most valuable lesson to be learned is that data collected from the human body tends to be noisy and is rather non-deterministic. This occurs not only due to the nature of the human body, but also due to the various factors affecting the instrumentation used and the ambient environment. Therefore it is crucial to first understand the phenomenon being measured and all factors that may affect it in detail, then design the system in a unit test style of development. That is, divide the platform's hardware and control software into the smallest possible testable units (or components). Test the units and rigorously ensure their operation as intended, and then assemble the system as each unit is verified. After all, an algorithm is only as accurate as the signal it is operating on. If the signal is distorted or inaccurate, then the algorithm's accuracy will surely suffer.

6.2 Limitations and Future Work

Since there are two platforms presented in this dissertation, we will present the limitations of each one separately. Each system currently functions well enough for research purposes and is reproducible, however they all require further fine tuning in order to make them production ready should the reader be interested in doing so.

Although XTREMIS produces a clean signal and is capable of predicting finger movements when performing ballistic gestures with approximately 94% accuracy, however it is important to address several issues moving forward. The first is placing it into a more user-friendly form factor: currently it rests as a standalone unit away from the user. A better form factor would be integrating it into a sleeve to be worn by the user. Secondly, to increase usability it is important to explore the use of dry-surface electrodes instead of the current wet-gel silver/silver chloride ones. Dry-surface electrodes will create a noisier signal but eliminate the need to use conductive gel for every experiment, making it more suitable to users not trained in ECG/EMG/EEG data collection. Finally, the sampling rate currently destabilizes at sampling rates above 500Hz. We have shown that this is due to the ADS1299 chip's design, however it would be interesting to explore some post-hoc processing steps that may be implemented to overcome this issue and increase the SNR even further at higher frequencies.

SiCILIA predicts the user's clothing insulation accurately within a 0.08 RMSE, and in fact – as shown in Chapter 3 – outperforms its competitor SPOT [25, 24] in both function and form. However, the performance is still dependent on the emissivity of the clothing worn. If the emissivity is too high (i.e. the occupant is wearing polyester or a thermally reflective material) then the perceived clothing surface temperature will be too close to the ambient temperature and the clothing insulation calculations will not be accurate. As such, there are a few more problems to solve in the future. First, SiCILIA should be capable of tracking occupants constantly in order to always be ready to take a measurement. Second, it must be able to differentiate between different object temperatures. For example, it must be able to differentiate between clothing, skin, and miscellaneous objects (such as a hot coffee mug). Lastly, SiCILIA should also be able to differentiate between good data and bad data. That is, whether or not the clo value obtained is reasonable in contrast to the ambient temperature or not.

Finally, CRONOS reduces the drift between sensors by approximately 97%, however it is not a fully automated system. Therefore it would be interesting to add more intelligence to the framework to detect the signal modality and the activity type so as to tailor the signal integrity checks accordingly. Additionally, it would be interesting to explore other sensing modalities and exploit other omnipresent components within their data. Indeed, CRONOS can be a flexible tool where users simply provide classifiers or templates for event recognition to receive a synchronized dataset.

Chapter 7

Appendix

7.1 Research Ethics

Research using XTREMIS (Project Number: 2016149) and SiCILIA (Project Number: 2014213) has been reviewed and approved by the McMaster University Research Ethics Board .

7.2 Subject Data

Table 7.1 shows the meta-data of each subject that participated in the experiments for SiCILIA. Whereas Table 7.2 presents the meta-data of each subject that participated int he experiments for XTREMIS. Note that the differing information provided between the two tables is due to the different MREB application requirements for the two systems.

Table 7.1: SiCILIA Subject Information

Subject	**Age**	**Height(cm)**	**Weight (kg)**	**BMI**	**Gender**
Subject 1	25	188cm	132	37.5	Male
Subject 2	25	173cm	65	22.7	Male
Subject 3	28	177cm	63	20.1	Male
Subject 4	26	188cm	108	30.6	Male
Subject 5	59	178cm	104	29.4	Male
Subject 6	28	162cm	49	18.7	Female
Subject 7	28	162cm	52	19.6	Female
Subject 8	20	163cm	77	29.0	Female
Subject 9	52	163cm	63	23.7	Female
Subject 10	27	170cm	52	18	Female

Table 7.2: XTREMIS Subject Information

Subject	**WPM Range**	**Gender**
Subject 1	25	Male
Subject 2	23	Male
Subject 3	24	Male
Subject 4	29	Male
Subject 5	28	Female
Subject 6	21	Female
Subject 7	24	Female
Subject 8	23	Female

Bibliography

[1] K. S. Abhishek, L. C. F. Qubeley, and D. Ho. 2016. Glove-based hand gesture recognition sign language translator using capacitive touch sensor. In *2016 IEEE International Conference on Electron Devices and Solid-State Circuits (EDSSC)*. 334–337. `https://doi.org/10.1109/EDSSC.2016.7785276`

[2] Eknoyan G AdolpheQuetelet. 2008. The average man and indices of obesity. *Nephrol Dial Transplant* 23 (2008), 47–51.

[3] Anand Asokan, Allan Joseph Pothen, and Raj Krishnan Vijayaraj. 2016. ARMatronA wearable gesture recognition glove: For control of robotic devices in disaster management and human Rehabilitation. In *Robotics and Automation for Humanitarian Applications (RAHA), 2016 International Conference on*. IEEE, 1–5.

[4] Alex Beltran, Varick L. Erickson, and Alberto E. Cerpa. 2013. ThermoSense: Occupancy Thermal Based Sensing for HVAC Control. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys'13)*. ACM, New York, NY, USA, Article 11, 8 pages. `https://doi.org/10.1145/2528282.2528301`

[5] Terrell R Bennett, Nicholas Gans, and Roozbeh Jafari. 2015. A data-driven synchronization technique for cyber-physical systems. In *Proceedings of the Second International Workshop on the Swarm at the Edge of the Cloud*. ACM, 49–54.

[6] Terrell R Bennett, Nicholas Gans, and Roozbeh Jafari. 2015. Multi-sensor data-driven: synchronization using wearable sensors. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*. ACM, 113–116.

[7] Terrell R Bennett, Nicholas Gans, and Roozbeh Jafari. 2017. Data-Driven Synchronization for Internet-of-Things Systems. *ACM Transactions on Embedded Computing Systems (TECS)* 16, 3 (2017), 69.

[8] John Bligh. 1975. Physiological Responses to Heat. (1975), 143–164. `https://doi.org/10.1007/978-1-4684-0760-0_7`

[9] Federico M Butera. 1998. Principles of thermal comfort. *Renewable and Sustainable Energy Reviews* 2, 1 (1998), 39–66.

[10] BioSemi B.V. 2017. BioSemi ActiveTwo Multi-modal data acquisition system with active electrodes. `https://biosemi.com`. (2017). Accessed: 2017-07-31.

[11] Natural Resources Canada. 2014. HVAC & Energy Systems. (2014). `http://canmetenergy.nrcan.gc.ca/buildings-communities/hvac-energy/908`.

[12] Xun Chen and Z Jane Wang. 2013. Pattern recognition of number gestures based on a wireless surface EMG system. *Biomedical Signal Processing and Control* 8, 2 (2013), 184–192.

[13] Richard J De Dear, Gail Schiller Brager, James Reardon, Fergus Nicol,

et al. 1998. Developing an adaptive model of thermal comfort and preference/discussion. *ASHRAE transactions* 104 (1998), 145.

[14] Travis Deyle, Szabolcs Palinko, Erika Shehan Poole, and Thad Starner. 2007. Hambone: A bio-acoustic gesture interface. In *Wearable Computers, 2007 11th IEEE International Symposium on*. IEEE, 3–10.

[15] Alexander Ekimov and James M Sabatier. 2006. Vibration and sound signatures of human footsteps in buildings. *The Journal of the Acoustical Society of America* 118, 3 (2006), 2021–768.

[16] J. Elson and D. Estrin. 2001. Time synchronization for wireless sensor networks. In *Proceedings 15th International Parallel and Distributed Processing Symposium. IPDPS 2001.* 1965–1970. `https://doi.org/10.1109/IPDPS.2001.925191`

[17] MJM English. 2001. Physical principles of heat transfer. *Current Anaesthesia & Critical Care* 12, 2 (2001), 66–71.

[18] PO Fanger. 1986. Thermal environment-Human requirements. *Environmentalist* 6, 4 (1986), 275–278.

[19] B Farnworth. 1983. Mechanisms of heat flow through clothing insulation. *Textile Research Journal* 53, 12 (1983), 717–725.

[20] K.A. Farry, I.D. Walker, and R.G. Baraniuk. 1996. Myoelectric teleoperation of a complex robotic hand. In *Robotics and Automation, IEEE Transactions on*, Vol. 12. 775–788. `https://doi.org/10.1109/70.538982`

[21] Antonio Fratini, Paolo Bifulco, Mario Cesarelli, Maria Romano, Giulio Pasquariello, Antonio La Gatta, and Gaetano Gargiulo. 2008. Muscle movement and electrodes motion artifact during vibration treatment. In *14th Nordic-Baltic Conference on Biomedical Engineering and Medical Physics*. Springer, 103–106.

[22] Lex Fridman, Daniel E Brown, William Angell, Irman Abdić, Bryan Reimer, and Hae Young Noh. 2016. Automated synchronization of driving data using vibration and steering events. *Pattern Recognition Letters* 75 (2016), 9–15.

[23] A Pharo Gagge, JAJ Stolwijk, and JD Hardy. 1967. Comfort and thermal sensations and associated physiological responses at various ambient temperatures. *Environmental research* 1, 1 (1967), 1–20.

[24] Peter Xiang Gao and S. Keshav. 2013. Optimal Personal Comfort Management Using SPOT+. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys'13)*. ACM, New York, NY, USA, Article 22, 8 pages. `https://doi.org/10.1145/2528282.2528297`

[25] Peter Xiang Gao and S. Keshav. 2013. SPOT: A Smart Personalized Office Thermal Control System. In *Proceedings of the Fourth International Conference on Future Energy Systems (e-Energy '13)*. ACM, New York, NY, USA, 237–246. `https://doi.org/10.1145/2487166.2487193`

[26] C. Gomez-Otero, R. Martinez, and J. Caffarel. 2012. ClimApp: A novel approach of an intelligent HVAC control system. In *Information Systems and Technologies (CISTI), 2012 7th Iberian Conference on*. 1–6.

[27] R. A. R. C. Gopura, K. Kiguchi, and Y. Li. 2009. SUEFUL-7: A 7DOF upper-limb exoskeleton robot with muscle-model-oriented EMG-based control. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1126–1131. `https://doi.org/10.1109/IROS.2009.5353935`

[28] Henry Gray and Charles Mayo Goss. 1878. *Anatomy of the human body*. Vol. 8. Lea & Febiger.

[29] Sidhant Gupta, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. Soundwave: using the doppler effect to sense gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1911–1914.

[30] Taeyoung Han and Linjie Huang. 2004. A model for relating a thermal comfort scale to EHT comfort index. (2004), 59–68.

[31] ASHRAE Fundamentals Handbook. 2009. *American society of heating, refrigerating and air-conditioning engineers.*

[32] ASHRAE Fundamentals Handbook. 2010. *American society of heating, refrigerating and air-conditioning engineers.*

[33] Fredric J Harris. 1978. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE* 66, 1 (1978), 51–83.

[34] George Havenith, Ingvar Holmér, and Ken Parsons. 2002. Personal factors in thermal comfort assessment: clothing properties and metabolic heat production. *Energy and Buildings* 34, 6 (2002), 581–591.

[35] S. Hijikata, K. Terabayashi, and K. Umeda. 2009. A simple indoor self-localization system using infrared LEDs. In *Networked Sensing Systems (INSS),*

2009 Sixth International Conference on. 1–7. `https://doi.org/10.1109/INSS.2009.5409955`

[36] Alan L Hodgkin and Andrew F Huxley. 1952. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology* 117, 4 (1952), 500–544.

[37] John R Howell, Robert Siegel, and M Pinar Menguc. 2011. *Thermal radiation heat transfer.* CRC press.

[38] Piccioli Alberto Moon Dustin Hoyt Tyler, Schiavon Stefano and Steinfeld Kyle. 2013. CBE Thermal Comfort Tool. Center for the Built Environment, University of California Berkeley. (2013). `http://cbe.berkeley.edu/comforttool/`

[39] Bernard Hudgins, Philip Parker, and Robert N Scott. 1993. A new strategy for multifunction myoelectric control. *Biomedical Engineering, IEEE Transactions on* 40, 1 (1993), 82–94.

[40] Charlie Huizenga, Hui Zhang, Edward Arens, and Danni Wang. 2004. Skin and core temperature response to partial-and whole-body heating and cooling. *Journal of Thermal Biology* 29, 7 (2004), 549–558.

[41] Thalmic Labs Inc. 2017. Myo Gesture Control Armband - Wearable Technology by Thalmic Labs. `https://www.myo.com`. (2017). Accessed: 2017-03-30.

[42] Texas Instruments. 2017. *ADS1299-x Low-Noise, 4-, 6-, 8-Channel, 24-Bit, Analog-to-Digital Converter for EEG and Biopotential Measurements.* Rev.C.

[43] Open Brain Computer Interface. 2017. OpenBCI - Biosensing for Everyone. `https://http://openbci.com/`. (2017).

[44] Y Itoh, H Uematsu, F Nogata, T Nemoto, A Inamori, K Koide, and H Matsuura. 2007. Finger curvature movement recognition interface technique using SEMG signals. *Journal of Achievements in Materials and Manufacturing Engineering* 23, 2 (2007), 43–46.

[45] Tianjian Ji et al. 2005. Frequency and velocity of people walking. *Structural Engineer* 84, 3 (2005), 36–40.

[46] Feng Jiang, Cuihua Wang, Yang Gao, Shen Wu, and Debin Zhao. 2015. Discriminating features learning in hand gesture classification. *Computer Vision, IET* 9, 5 (2015), 673–680.

[47] Gary Kamen and Electromyographic Kinesiology. 2004. Research methods in biomechanics. *Champaign, IL, Human Kinetics Publ* (2004).

[48] Gurmanik Kaur, Ajat Shatru Arora, and VK Jain. 2009. Comparison of the techniques used for segmentation of EMG signals. In *Proceedings of the 11th WSEAS international conference on Mathematical and computational methods in science and engineering*. World Scientific and Engineering Academy and Society (WSEAS), 124–129.

[49] J. H. Kim, N. D. Thang, and T. S. Kim. 2009. 3-D hand motion tracking and gesture recognition using a data glove. In *2009 IEEE International Symposium on Industrial Electronics*. 1013–1018. `https://doi.org/10.1109/ISIE.2009.5221998`

[50] C. Knapp and G. Carter. 1976. The generalized correlation method for estimation of time delay. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 24, 4 (Aug 1976), 320–327. `https://doi.org/10.1109/TASSP.1976.1162830`

[51] Kenneth R. Koehler. 1996. *Body Temperature Regulation*. University of Cincinnati Raymond Walters College.

[52] G. Kondo, R. Kato, H. Yokoi, and T. Arai. 2010. Classification of individual finger motions hybridizing electromyogram in transient and converged states. *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (May 2010), 2909–2915. `https://doi.org/10.1109/ROBOT.2010.5509493`

[53] Richard A Kronmal. 1993. Spurious correlation and the fallacy of the ratio standard revisited. *Journal of the Royal Statistical Society* (1993), 379–392.

[54] Abraham Hang-yat Lam, Yi Yuan, and Dan Wang. 2014. An occupant-participatory approach for thermal comfort enhancement and energy conservation in buildings. In *Proceedings of the 5th international conference on Future energy systems*. ACM, 133–143.

[55] Jun Liu, Zhaohui Wang, Michael Zuba, Zheng Peng, Jun-Hong Cui, and Shengli Zhou. 2014. DA-Sync: A Doppler-assisted time-synchronization scheme for mobile underwater sensor networks. *IEEE Transactions on Mobile Computing* 13, 3 (2014), 582–595.

[56] Martin Lukac, Paul Davis, Robert Clayton, and Deborah Estrin. 2009. Recovering temporal integrity with data driven time synchronization. In *Proceedings*

of the 2009 International Conference on Information Processing in Sensor Networks. IEEE Computer Society, 61–72.

[57] T. Mantecn, A. Mantecn, C. R. del Blanco, F. Jaureguizar, and N. Garca. 2015. Hand-gesture-based human-machine interface system using Compressive Sensing. In *2015 International Symposium on Consumer Electronics (ISCE)*. 1–2. `https://doi.org/10.1109/ISCE.2015.7177828`

[58] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. 2004. The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM, 39–49.

[59] Elizabeth A McCullough, Byron W Jones, and J Huck. 1985. A comprehensive data base for estimating clothing insulation. *ASHRAE Transactionsq* 91, 2 (1985), 29–47.

[60] R Merletti. 1999. *ISEK Standards of Reporting EMG Data.*

[61] David L Mills. 1997. Computer network time synchronization. *Report Dagstuhl Seminar on Time Services Schloß Dagstuhl, March* 11 (1997), 332.

[62] Sushmita Mitra and Tinku Acharya. 2007. Gesture recognition: A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 37, 3 (2007), 311–324.

[63] Michele Pla Mobarak, Manuel Juan Gutierrez Salgado, Roberto Munoz Guerrero, and Valérie Louis Dorr. 2014. Transient state analysis of the multichannel EMG signal using Hjorth's parameters for identication of hand movements. In

International Multi-Conference on Computing in the Global Information Technology, ICCGI 2014.

[64] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingerio: Using active sonar for fine-grained finger tracking. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 1515–1525.

[65] Sujay Narayana, R. Venkatesha Prasad, Vijay S. Rao, T. V. Prabhakar, Sripad S. Kowshik, and Madhuri Sheethala Iyer. 2015. PIR Sensors: Characterization and Novel Localization Technique. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks (IPSN '15)*. ACM, New York, NY, USA, 142–153. `https://doi.org/10.1145/2737095.2742561`

[66] R. Tugrul Ogulata. 2007. The Effect of Thermal Insulation of Clothing on Human Thermal Comfort. *FIBRES & TEXTILES in Eastern Europe* 15, 2 (2007), 67–72.

[67] A. Virgílio M. Oliveira, Adélio R. Gaspar, and Divo A. Quintela. 2008. Measurements of clothing insulation with a thermal manikin operating under the thermal comfort regulation mode: comparative analysis of the calculation methods. Vol. 104. 679–688. `https://doi.org/10.1007/s00421-008-0824-5`

[68] Mohammadreza Asghari Oskoei, Huosheng Hu, and John Q Gan. 2008. Manifestation of fatigue in myoelectric signals of dynamic contractions produced during playing PC games. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*. IEEE, 315–318.

[69] Dawei Pan, Abraham Hang-yat Lam, and Dan Wang. 2013. Carrying my environment with me in IOT-enhanced smart buildings. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services.* ACM, 521–522.

[70] Panasonic Corporation. 2016. Infrared Array Sensor Grid-EYE. (2016). `http://industrial.panasonic.com/ww/products/sensors/built-in-sensors/grid-eye` [Online; accessed February 19, 2016].

[71] Fabio Pareschi, Pierluigi Albertini, Giovanni Frattini, Mauro Mangia, Riccardo Rovatti, and Gianluca Setti. 2016. Hardware-algorithms co-design and implementation of an analog-to-information converter for biosignals based on compressed sensing. In *IEEE transactions on biomedical circuits and systems*, Vol. 10. IEEE, 149–162.

[72] Luis Pérez-Lombard, José Ortiz, and Christine Pout. 2008. A review on buildings energy consumption information. *Energy and buildings* 40, 3 (2008), 394—398.

[73] Angkoon Phinyomark, Pornchai Phukpattaranont, and Chusak Limsakul. 2012. Feature reduction and selection for EMG signal classification. *Expert Systems with Applications* 39, 8 (2012), 7420–7431.

[74] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking.* ACM, 27–38.

[75] Chotirat Ann Ratanamahatana and Eamonn Keogh. 2004. Everything you know about dynamic time warping is wrong. *Third Workshop on Mining Temporal and Sequential Data* (2004).

[76] MBI Reaz, MS Hussain, and Faisal Mohd-Yasin. 2006. Techniques of EMG signal analysis: detection, processing, classification and applications. *Biological procedures online* 8, 1 (2006), 11–35.

[77] Carrie A Redlich, Judy Sparer, and Mark R Cullen. 1997. Sick-building syndrome. *The Lancet* 349, 9057 (1997), 1013–1016.

[78] Gordon Robertson, Graham Caldwell, Joseph Hamill, Gary Kamen, and Saunders Whittlesey. 2013. *Research methods in biomechanics, 2E.* Human Kinetics.

[79] Kay Römer. 2001. Time synchronization in ad hoc networks. In *Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*. ACM, 173–182.

[80] R. Rosenberg. 1998. The biofeedback pointer: EMG control of a two dimensional pointer. *Second International Symposium on Wearable Computers* (Oct 1998), 162–163. `https://doi.org/10.1109/ISWC.1998.729546`

[81] J Rugh and V Hovland. 2003. National and world fuel savings and CO2 emission reductions by increasing vehicle air conditioning COP. In *Proceedings from the 2003 Alternate Refrigerant Systems Symposium in Phoenix, AZ.*

[82] John P Rugh, Robert B Farrington, Desikan Bharathan, Andreas Vlahinos, Richard Burke, Charlie Huizenga, and Hui Zhang. 2004. Predicting human

thermal comfort in a transient nonuniform thermal environment. *European journal of applied physiology* 92, 6 (2004), 721–727.

[83] Saeid Sanei and Jonathon A Chambers. 2013. *EEG signal processing.* John Wiley & Sons.

[84] Ala Shaabana, Joey Legere, Jun Li, Rong Zheng, Martin v. Mohrenschildt, and Judith Shedden. 2017. XTREMIS Repository. `https://github.com/shibshib/XTREMIS_CAPSTONE`. (2017).

[85] Ala Shaabana, Joey Legere, Jun Li, Rong Zheng, Martin v. Mohrenschildt, and Judy Shedden. 2018. XTREMIS: A Portable Biomedical Sensing Platform (Submitted). *ACM Transactions on Embedded Computing Systems* (2018).

[86] Ala Shaabana and Rong Zheng. 2017. CRONOS Repository. `https://github.com/shibshib/SenSync`. (2017).

[87] Ala Shaabana and Rong Zheng. 2018. CRONOS: A Post-hoc Data Driven Multi-Sensor Synchronization Approach (Submitted). In *Proceedings of the 17th International Conference on Information Processing in Sensor Networks*. ACM, 1–13.

[88] A. Shaabana, R. Zheng, J. Legere, and M. V. Mohrenschildt. 2017. Finger Movement Recognition During Ballistic Movements Using Electromyography. *2017 IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE)* (July 2017), 302–311. `https://doi.org/10.1109/CHASE.2017.113`

[89] Ala Shaabana, Rong Zheng, Joey Legere, and Martin v. Mohrenschildt. 2017. Finger movement recognition during ballistic movements using electromyography. In *Proceedings of the Second IEEE/ACM conference on Connected health: Applications, Systems and Engineering technologies*. IEEE, 302–311.

[90] Ala Shaabana, Rong Zheng, and Zhipeng Xu. 2015. SiCILIA: A Smart Sensor System for Clothing Insulation Inference. In *2015 IEEE Global Communications Conference (GLOBECOM)*. 1–6.

[91] Ala Shaabana, Rong Zheng, and Zhipeng Xu. 2015. SiCILIA: A smart sensor system for clothing insulation inference using heat exchange. *Proceedings of the 14th International Conference on Information Processing in Sensor Networks* (2015), 362–363.

[92] Ala Shaabana, Rong Zheng, and Zhipeng Xu. 2017. SiCILIA: Inferring Clothing Insulation Levels using Mechanisms of Heat Transfer (IN PRESS). *ACM Transactions on Sensor Networks* (2017).

[93] Ala Shaabana, Rong Zheng, and Xu Zhipeng. 2017. SiCILIA Repository. `https://github.com/shibshib/SiCILIA`. (2017).

[94] Yifang Si, Junqi Yu, Nan Wang, Xisheng Ding, Teng Guo, and Longfei Yuan. 2016. Indoor comfort assessment of objective and subjective information by fusion and fuzzy inference decision. *Intelligent Buildings International* 8, 4 (2016), 234–245.

[95] Roberto Solis, Vivek S Borkar, and PR Kumar. 2006. A new distributed time

synchronization protocol for multihop wireless networks. In *Decision and Control, 2006 45th IEEE Conference on.* IEEE, 2734–2739.

[96] B. Song, H. Choi, and H. S. Lee. 2008. Surveillance Tracking System Using Passive Infrared Motion Sensors in Wireless Sensor Network. *2008 International Conference on Information Networking* (Jan 2008), 1–5. `https://doi.org/10.1109/ICOIN.2008.4472790`

[97] Mani Srivastava, Tarek Abdelzaher, and Boleslaw Szymanski. 2012. Human-centric sensing. *Philosophical Transactions of The Royal Society A* 370, 1958 (2012), 176–197.

[98] H Graham Stack. 1962. Muscle function in the fingers. *Journal of Bone & Joint Surgery, British Volume* 44, 4 (1962), 899–909.

[99] Thad Starner, Joshua Weaver, and Alex Pentland. 1998. Real-time american sign language recognition using desk and wearable computer based video. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20. IEEE, 1371–1375.

[100] Richard S Taylor. 2010. Use of body mass index for monitoring growth and obesity. *Paediatrics & Child Health* 15, 5 (2010), 258–258.

[101] Azusa Uematsu, Koh Inoue, Hiroaki Hobara, Hirofumi Kobayashi, Yuki Iwamoto, Tibor Hortobgyi, and Shuji Suzuki. 2011. Preferred step frequency minimizes veering during natural human walking. *Neuroscience Letters* 505, 3 (2011), 291 – 293. `https://doi.org/10.1016/j.neulet.2011.10.057`

[102] Jue Wang, Deepak Vasisht, and Dina Katabi. 2014. RF-IDraw: virtual touch screen in the air using RF signals. *ACM SIGCOMM Computer Communication Review* 44, 4 (2014), 235–246.

[103] John Webster. 2009. *Medical instrumentation: application and design*. John Wiley & Sons.

[104] Peter Welch. 1967. The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. In *IEEE Transactions on audio and electroacoustics*, Vol. 15. IEEE, 70–73.

[105] Jonathan CK Wells. 2001. A critique of the expression of paediatric body composition data. *Archives of disease in childhood* 85, 1 (2001), 67–72.

[106] Kevin R Wheeler. 2003. Device control using gestures sensed from EMG. *Proceedings of the 2003 IEEE International Workshop on Soft Computing in Industrial Applications, 2003 (SMCia/03)* (2003), 21–26.

[107] Sheena Wilson and Alan Hedge. 1987. *The office environment survey: a study of building sickness*. Building Use Studies Limited.

[108] Ernst Wit, Edwin van den Heuvel, and Jan-Willem Romeijn. 2012. All models are wrong...: an introduction to model uncertainty. *Statistica Neerlandica* 66, 3 (2012), 217–236.

[109] Yik-Chung Wu, Qasim Chaudhari, and Erchin Serpedin. 2011. Clock synchronization of wireless sensor networks. *IEEE Signal Processing Magazine* 28, 1 (2011), 124–138.

[110] Yong Yao and Johannes Gehrke. 2002. The cougar approach to in-network query processing in sensor networks. *ACM Sigmod Record* 31, 3 (2002), 9–18.

[111] Yi Yuan, Dawei Pan, Dan Wang, Xiaohua Xu, Yu Peng, Xiyuan Peng, and Peng-Jun Wan. 2013. A Study Towards Applying Thermal Inertia for Energy Conservation in Rooms. *ACM Transactions on Sensor Networks* 10, 1, Article 7 (Dec. 2013), 25 pages. `https://doi.org/10.1145/2529050`

[112] Hui Zhang. 2003. *Human thermal sensation and comfort in transient and non-uniform thermal environments.* Ph.D. Thesis.

[113] Xu Zhang, Xiang Chen, Wen-hui Wang, Ji-hai Yang, Vuokko Lantz, and Kong-qiao Wang. 2009. Hand gesture recognition and virtual game control based on 3D accelerometer and EMG sensors. In *Proceedings of the 14th international conference on Intelligent user interfaces.* ACM, 401–406.