

WEARABLE INTERFACES FOR SYMBOLIC COMMUNICATION BY WORKING DOGS

A Dissertation
Presented to
The Academic Faculty

By

Giancarlo Valentin

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in
Electrical and Computer Engineering



School of Electrical and Computer Engineering
Georgia Institute of Technology
May 2019

Copyright © 2019 by Giancarlo Valentin

WEARABLE INTERFACES FOR SYMBOLIC COMMUNICATION BY WORKING DOGS

Approved by:

Dr. Elliot Moore, Committee Chair
*Professor, School of Electrical and Computer
Engineering
Georgia Institute of Technology*

Dr. Omer Inan
*Professor, School of ECE
Georgia Institute of Technology*

Dr. Ayanna Howard, Co-advisor
*Professor, School of Electrical and Computer
Engineering
Georgia Institute of Technology*

Dr. Thad Starner
*Professor, School of Interactive Computing
Georgia Institute of Technology*

Dr. Melody M. Jackson, Co-advisor
*Professor, School of Interactive Computing
Georgia Institute of Technology*

Dr. Thomas Ploetz
*Professor, School of Interactive Computing
Georgia Institute of Technology*

Date Approved: February 28 2019

In loving memory of Sky...

ACKNOWLEDGMENTS

I would first like to thank my advisors Dr. Ayanna Howard and Dr. Melody Moore Jackson without whom none of my work would be possible. Similarly, I am indebted to my committee members Dr. Thad Starner, Dr. Thomas Ploetz, Dr. Omer Inan and Dr. Elliot Moore for their valuable time and input. I am also grateful to my other mentors and counselors during my time at Georgia Tech. They are Dr. Allen Robinson, Dr. Whit Smith, Dr. Gregory Abowd, Dr. Clint Zeagler, Dr. Hae Won Park and Dr. Lavonda Brown.

I would also like recognize Kevin Pham and James Steinberg (ECE) and Scott Gilliland and Jay Zuerndorfer (GVU) for sharing their immense knowledge of certain hardware aspects that helped improve my work. Regarding foundations of communication, Dr. Clara Mancini brought to my attention the importance of distinguishing between symbols and other types signs used for communication.

The first generation of tangible interfaces benefited from help from Wendy Blount, Sarah J. Eiring, Kevin Martin and Adil Delawalla. The second generation of attachments received great improvements from Lily Burkeen, Larry Freil and Ceara Byrne.

On the pattern recognition aspects Dr. Daniel Kohlsdorf was a valuable technical mentor and brought to my attention the importance of distinguishing between activity and gesture recognition. I understood the importance of segmentation in continuous recognition from Dr. Thomas Ploetz's tutorial at Ubicomp 2015. I also learned a great deal from Aman Parmani's presentations on gesture creation for humans. Ryan Kerwin, Ivan Walker and Joelle Alcaidinho were amazing in providing valuable feedback during each step of this work. Ryan wrote scripts to transfer ELAN annotations to raw data and later expanded them to produce input files into the WEKA toolkit.

On the dog training side, Barbara Currier was instrumental in her guidance and this project would not be possible without her. Vince Martin, Gaby Gammans, and Wallis Brosnan provided guide dog, search and rescue and assistance dog scenarios that served

us extremely well. Wallis could be considered our first user and her feedback proved essential. Rob Turner and Zach Bryant of GTPD had a similar role instructing us about law-enforcement scenarios. Similarly, I want to thank our human participants Margo Gathright-Dietrich, Candace Atchison, Peggy Donato, Ninette Franz, and Alyssa Eidbo.

This work was received financial support from the National Science Foundation (1320690), the Georgia Tech GVU, and the Georgia Tech Wearable Computing Center.

I am personally grateful for support provided by Intel Corporation through the Intel Scholars program, as well as the GEM Consortium through the GEM fellowship. For these fantastic experiences, I am in debt to Lisa Kleinman, Jamie Sherman, Lama Nachman, Rahul Shah, and Shameeka Emanuel.

Last but not least, I thank our main protagonists and canine collaborators. They were Sky, Blitz, Schubert, Manolo, Stormy, Miley, Koda and many others who made the past years a great adventure.

TABLE OF CONTENTS

ACKNOWLEDGMENTS	iv
LIST OF TABLES	x
LIST OF FIGURES	xii
SUMMARY	xv
CHAPTER 1 INTRODUCTION	1
1.1 Definition of working dogs	1
1.2 Barriers to communication	2
1.3 Theory of communication	3
1.4 Thesis statement	5
CHAPTER 2 LITERATURE SURVEY	6
2.1 Canine communication	6
2.1.1 Alerting on environmental cues	6
2.1.2 Canine-machine interaction	7
2.1.3 Augmented working dog communication	8
2.1.4 Symbolic canine-to-human communication	8
2.2 Automated sequence recognition	9
2.2.1 Recognition for human-machine interaction	10
2.2.1.1 Sequence recognition for user interfaces	11
2.2.1.2 Direct manipulation interfaces	12
2.2.1.3 Automated gesture recognition for user interfaces	13
2.2.1.4 Usability of gesture interfaces	14
2.2.1.5 Automated gesture recognition for always-on interactions	16
2.2.2 Automated activity recognition	17
2.2.2.1 Human activity recognition	17
2.2.2.2 Canine activity (and posture) recognition	19
2.3 Comparison and limits of sequence recognition	19
2.3.1 Objectives of supervised recognition	19
2.3.2 Limits to canine gesture recognition	21
CHAPTER 3 WEARABLE TANGIBLE INTERFACES	25
3.1 Introduction	25
3.2 Resistive bite interfaces	26
3.2.1 Bite detection mechanism	27
3.3 Resistive tug interfaces	27
3.3.1 Tug detection mechanism	28
3.4 Touch interfaces	29
3.4.1 Touch detection mechanism	29

3.5	Protocol and participants for wearable tangible interfaces	30
3.6	Method	30
3.7	Performance metrics	32
3.8	Results	32
3.9	Discussion	32
3.9.1	User feedback	33
CHAPTER 4	WEARABLE GESTURE INTERFACES	35
4.1	Introduction	35
4.2	Harness-based offline system	36
4.2.1	Performance and evaluation	37
4.2.2	Lessons learned	38
4.3	On-board collar-based system	39
4.3.1	Participants	39
4.3.2	System and equipment	40
4.3.3	Movements of interest	41
4.3.4	Gesture training protocol	41
4.3.4.1	Identification of horizontal movements	42
4.3.4.2	Excluded basic movement types	43
4.3.4.3	Identification of rotational movements	44
4.3.4.4	Provisional solutions	44
4.3.4.5	Movements selected	45
4.3.4.6	Identification of gesture sequences	46
4.3.5	Experimental procedure	46
4.3.5.1	Performance metrics	47
4.3.5.2	System accuracy metrics for sequences	48
4.3.5.3	Repetition experiment	49
4.3.5.4	False positives in urban environment	49
4.3.5.5	False positives in open environment	50
4.3.6	Results	50
4.3.6.1	Repetition experiment	50
4.3.6.2	False positive experiment	53
4.3.7	Discussion	53
4.3.7.1	Experiment and system improvements	53
4.4	Conclusion	54
CHAPTER 5	GESTURE CONCEPTION, SELECTION AND DEFINITION	56
5.1	Introduction	56
5.2	Gesture selection constraints	56
5.2.1	Seven requirements	57
5.2.2	Plotting the requirements	58
5.2.3	Separating the constraints	59
5.2.3.1	Gesture selection	59
5.3	Protocol and participants for false positive study	61
5.3.1	Participants	62

5.3.2	Dog training protocol	62
5.4	System and equipment	63
5.5	Detection and classification considerations	63
5.6	Inertial data annotation	64
5.6.1	Orientation correction	65
5.6.1.1	Obtaining gravity from acceleration	66
5.6.1.2	Variance of three-axis norm	67
5.6.2	Event segmentation	68
5.6.2.1	Distance metric	69
5.6.2.2	Parameter tuning	69
5.7	Discussion	70
5.7.1	Types of false positives	72
5.8	Challenges of the gesture set	72
5.9	Conclusion	73
CHAPTER 6 RECOGNITION OF MOVEMENTS IN CONSTRAINED ENVIRONMENTS		75
6.1	Introduction	75
6.2	Participants	76
6.3	Equipment	76
6.4	Collecting training data	77
6.5	Detection and classification	78
6.6	Classifier and feature selection	78
6.7	Evaluation	79
6.8	Results	79
6.8.1	Support vector machine	80
6.8.2	Stochastic gradient descent	80
6.8.3	K nearest neighbors	81
6.8.4	Decision trees	82
6.8.5	Random forests	82
6.9	Discussion	83
6.10	Conclusion	84
CHAPTER 7 WEARABLE GESTURE RECOGNITION SYSTEM		85
7.1	Introduction	85
7.2	Participants	85
7.3	Equipment	86
7.4	Data collection	87
7.4.1	Selection of objects to alert on	87
7.4.2	Training to alert at sign	87
7.4.3	Training to discriminate between sign symbols	90
7.4.4	Scent training	91
7.4.5	Discriminating between gesture movements	91
7.4.6	Data collection resolution	92
7.4.7	Protocol	93

7.4.8	Remaining difficulties	93
7.5	Data annotation	94
7.5.1	Types of annotations	95
7.5.2	Synchronization of video and inertial data	96
7.5.3	Inertial data annotation	97
7.5.3.1	Multi-axis single-sample labels	98
7.5.3.2	Single-axis sample labels	98
7.5.3.3	Single movement annotations	100
7.5.3.4	Single movement segmentation	103
7.5.3.5	Combined movement grouping	104
7.5.3.6	Combined movement annotation	105
7.5.3.7	Other combined movement annotations	107
7.5.3.8	Annotations for this study	109
7.5.4	Video annotation	109
7.6	Segmentation and feature selection	111
7.6.1	Single movement segmentation	112
7.6.2	Feature selection	112
7.6.2.1	Single movement features	114
7.6.2.2	Combined movement features	115
7.7	Results and classification	115
7.7.1	Data sets	116
7.7.2	Results of offline evaluation	116
7.8	Discussion	117
7.8.1	Comparison of annotation and classification	118
7.9	Conclusion	121
CHAPTER 8 WEARABLE GESTURE SYSTEM EVALUATION		124
8.1	Introduction	124
8.2	Participants	124
8.3	Inertial and video data annotation	124
8.4	Experiment setup	124
8.5	Results	125
8.5.1	Results of classification of all combined movements	126
8.5.2	Results of classification of candidate gestures	126
8.5.3	Breed-independent analysis	127
8.6	Discussion	128
8.7	Future work	129
8.8	Conclusion	131
REFERENCES		132

LIST OF TABLES

Table 1	Reported accuracy of Wiimote movements.	14
Table 2	Types of movements and their functions.	15
Table 3	Differences between automated recognition problems in related areas. . .	24
Table 4	Participant demographics for the tangible interface study.	31
Table 5	Definition of terms for each of the three performance metrics.	32
Table 6	Interface Detection Accuracy for each type of interface.	33
Table 7	Subject demographics. <i>Retriever cross</i> denotes a cross between labrador retriever and golden retriever.	39
Table 8	Summary of all the movements analyzed in this study.	46
Table 9	Definitions for performance metrics for the second study.	48
Table 10	Sequences depended on the correct detection of single gestures and compound gestures.	49
Table 11	Results for a new metric, sequences II that only counts cases where the basic units were detected correctly.	49
Table 12	Cue Response Accuracy for each subject. Note that the training methods used were different for each one.	51
Table 13	System accuracy for S1. Sequences II analyzes the detection of gestures sequences by controlling for cases where the compound gesture should have been detected.	51
Table 14	System accuracy for S2. Sequences II analyzes the detection of gestures sequences by controlling for cases where the compound gesture should have been detected.	52
Table 15	System accuracy for S3. In this case, gz and gx were combined.	53
Table 16	False positives experiment. The up and down gestures triggered much more so than the other, while still being difficult to perform. For this reason they were not included in the repetition experiments.	54
Table 17	Definition of each gesture movement under consideration.	60
Table 18	Segmentation criteria determined empirically for movements sampled at 51.2 Hz where <i>theta</i> denotes the angle of movement (e.g., an ideal Spin is 360 °).	69

Table 19	Summarized results for each data set. Some dogs offered gesture movements more than four times.	71
Table 20	Preliminary analysis of combined data-sets through five-fold cross-validation.	79
Table 21	Leave-one-subject out cross validation F1-performance metric.	80
Table 22	Confusion matrix for the SVM classifier.	80
Table 23	Confusion matrix for the SGD classifier.	81
Table 24	Confusion matrix for the kNN classifier.	81
Table 25	Confusion matrix for the decision tree classifier.	82
Table 26	Confusion matrix for the random forest classifier.	82
Table 27	Participant information for the wearable gesture study.	85
Table 28	Definitions for single-axis movements according to gyroscope readings. .	98
Table 29	Summary of multi-axes labels and their names in this study.	100
Table 30	Summary of the set of multi-sample single movements defined for this experiment.	103
Table 31	Annotation parameters that can be set by the human.	106
Table 32	Summary of the set of triplet labels defined in this study.	109
Table 33	Inertial data annotation of gesture triplets for BC1 sessions.	110
Table 34	Inertial data annotation of gesture triplets for BC2 sessions.	111
Table 35	Full comparison of results for BC1 sessions.	121
Table 36	Full comparison of results for BC2 sessions.	122
Table 37	Participant table. For this study we used data on four subjects.	124
Table 38	Inertial and video annotation for BC3 sessions.	125
Table 39	Comparison of labels for BC3 sessions.	127
Table 40	Results of the retriever performing spin and twirl movements.	128

LIST OF FIGURES

Figure 1	The three classic types of signs studied in the field of semiotics.	4
Figure 2	Two types of signs dogs are known to perform. Our focus is strictly on symbolic signs.	4
Figure 3	Venn diagram illustrating the intersection of three fields of study relevant to canine-machine interaction.	6
Figure 4	Keys of the lexigram keyboard used by Rossi et al. during their experiments [1].	9
Figure 5	The Theremin, an early gesture-based system, is an electronic musical instrument controlled by the movements of a performer.	10
Figure 6	Two participants activating early iterations of each interface [2].	25
Figure 7	Construction of bite interface from a force sensitive resistor.	26
Figure 8	The oval-shaped bite interface improved results by inducing the bite force to be applied vertically.	27
Figure 9	Construction of four-sided bite interface from a force sensitive resistor.	27
Figure 10	Input configuration for bite and tug interfaces and activation graph for one session with the four-sided bite interface.	28
Figure 11	Detailed illustration of early versions of the instrumented harnesses and their components.	30
Figure 12	System diagram of a gesture-based communication system.	35
Figure 13	We used the Axivity accelerometer (circled) in this first experiment.	36
Figure 14	WAX9 inertial sensor attached to a dog collar by two rubber bands supported by an equally sized polyurethane foam piece.	40
Figure 15	The companion application has a visual representation of the basic movements for the purposes of training. Note that spin and twirl are not purely head-only gestures but also involve body movement.	42
Figure 16	Gyroscope measurements for left and right reach with a completely planar movement.	42
Figure 17	Basic scheme to detect gestures with gz representing the yaw axis of the gyroscope.	45

Figure 18	Finite state machine for detecting gesture-like sequences. Dog trainers refer to these sequences as behavior chains. Gesture movement i and gesture movement j represent names of two consecutive gestures	47
Figure 19	Assistance dog walks through sidewalk while wearing the instrumented collar.	50
Figure 20	Explosive detection canine performing the open environment experiment.	50
Figure 21	The use of a treat to lure the reach gestures caused problems with detection.	52
Figure 22	Ideal characteristics of gestures for detection. For canines, the ease of training aspects is of utmost importance.	58
Figure 23	Gesture constraints relative to single horizontal gestures, which serve as a baseline.	59
Figure 24	Requirements for ideal gestures separated between gesture requirements and system requirements.	60
Figure 25	One participant performs the twirl (top) and left reach (bottom) gesture. .	60
Figure 26	Participant wearing the instrumented Julius K9 harness and Shimmer 3 before a training session.	62
Figure 27	Comparison of a raw acceleration signal of a single gesture and the resulting gravity vector over time. We used these values to correct the orientation on gyroscope readings.	67
Figure 28	Variance norm in green overlaid over one amplified dimension of the raw signal for comparison.	68
Figure 29	Example of gesture templates used for comparison against streams of data.	70
Figure 30	Example result of a right and left reach detected or misclassified in data sets containing other movements.	71
Figure 31	Visual description of the movement recognition pipeline.	76
Figure 32	Subject illustrating how the Moto 360 was worn on the collar. This dog was not part of this study.	77
Figure 33	Final enclosure and orientation to hold the shimmer sensor parallel to the collar.	86
Figure 34	The final objects consisted of small scale traffic signs.	88
Figure 35	Onleash training performed indoors.	89

Figure 36	Visual description of the training protocol.	94
Figure 37	One participant being rewarded after completing one gesture.	95
Figure 38	Using the ELAN annotation toolkit to synchronize the streams of inertial and video data.	96
Figure 39	Example of a right reach gesture, the red, blue and green lines represent roll, pitch and yaw respectively.	99
Figure 40	In continuous recognition the human provides annotations but these are re-applied to individual frames or windows.	102
Figure 41	We devised a third approach, where the system applies the annotations programatically through human supervision.	104
Figure 42	Example of a dog performing one right reach gesture after another. . . .	107
Figure 43	Example of the feature vector for one multi-sample segment.	114
Figure 44	Example of the feature vector for a given triplet.	115
Figure 45	Cumulative Confusion matrix of SVM classifier for leave-one-session-out validation.	117
Figure 46	Cumulative confusion matrix of SGD classifier for leave-one-session-out validation.	118
Figure 47	Cumulative confusion matrix of nearest neighbors classifier for leave-one-session-out validation.	119
Figure 48	Cumulative confusion matrix of random forest classifier for leave-one-session-out validation.	120
Figure 49	Confusion matrix of random forest classifier for leave-one-subject-out validation.	126

SUMMARY

Working dogs are canines with one or more specific skills that enable them to perform essential tasks for humans. Unfortunately, the information they perceive often exceeds their ability to communicate it. After examining a set of low-communication scenarios, we identified three factors (perception, distance and context) that form barriers to communication between working dogs and humans. Next, we present solutions to decrease their effect using wearable technology. An early approach was to create wearable interfaces that a dog could activate through common abilities (biting, tugging or touching) to alert a human. The results were very encouraging and suggested working dogs could be trained to reliably activate on-body interfaces. We then considered how to gradually minimize the equipment and allow dogs to generate more than one alert per interface. Our second approach explored the automatic detection of gestures sensed from inertial motion on the collar to create the alerts for communication. From the first of these studies we discovered a set of often-conflicting requirements a gesture must meet to be successful for communication. We made these requirements explicit and examined a series of four gestures that could meet them by comparing their similarity against data of everyday movements. Finally, we developed a pipeline for annotating and classifying canine gesture-like movements. We annotated the most basic movements and showed how their basic features could be combined to form larger gestures for communication. The outcome of this research is the development of technologies for a wearable gesture system that allows symbolic communication between working dogs and humans despite differences in each one's perceptual abilities, distance and context.

CHAPTER 1

INTRODUCTION

1.1 Definition of working dogs

Working dogs are canines with one or more specific skills that enable them to perform essential tasks for humans. Working dogs that assist humans with disabilities are called assistance dogs. Other working dog occupations include field work, such as search and rescue (SAR) or explosive-detection.

The roles of working dogs continue to evolve alongside, and sometimes with, technology. For example, just like advances in semiconductor physics have led to increased capabilities in the fields of computing, sensing and automation, so have advances in the field of canine cognition augmented the possibilities for dog–human partnerships.

In this dissertation, we study and develop wearable systems that can minimize factors inhibiting communication between working dogs and the humans they assist. Because we want to minimize these factors we refer to them as metaphorical barriers to communication that we must overcome. In the present chapter, we begin by identifying these barriers to canine–human communication (Section 1.2).

In Chapter 3 we detail our earliest experiments creating new communication channels using wearable interfaces that dogs could activate to communicate discrete information to a human. In Chapter 4 we first considered the use of inertially-sensed gestures for communication and analyze the trade-offs between each of seven necessary criteria we determined for successful gesture movements. Next, we recorded four candidate dog gestures that had not failed these criteria and estimated their propensity for false positives (sensitivity) by comparing the similarity between inertial templates of each one and inertial recordings of everyday movements (Chapter 5 Section 5.2.3.1).

We used the findings from these studies to formulate the attributes of each gesture movement relative to a set of seven requirements necessary in gestures for canine–human

communication (Chapter 5). We conclude by assembling a wearable gesture recognition system (Chapter 7) and performing an evaluation of its recognition performance (Chapter 8).

1.2 Barriers to communication

We began our studies by conducting interviews with human companions of guide dogs, assistance dogs, and search and rescue dogs. These interviews suggested the information perceived by working dogs often exceeds their ability to communicate with humans [3]. There are multiple factors that inhibit this communication and often improving one factor can worsen another. To ensure awareness of these trade-offs, and to address them as a whole, we define and classify them into three categories.

- Perceptual barriers
- Distance barriers
- Contextual barriers

Perceptual barriers are a result of dogs needing to communicate something they can sense but their human companions cannot. This type of barrier might be the result of a person's disability (e.g., visual or hearing impairment) or a human sensory limitation compared to canines (e.g., scent). Because human senses cannot perceive what the dog is sensing, the information *must* be communicated explicitly through the remaining available channels.

Distance barriers are present, for example, in canine-aided search and rescue, where the dogs' most commonly understood communication signals (i.e., barking, positioning, etc.) might be ineffective at distances beyond line of sight or hearing [4, 5].

Contextual barriers are manifest when service dogs must communicate with unfamiliar humans. For example, in case of health emergencies, such as the human companion having a seizure, some service dogs must alert other humans. Because their signaling behaviors

is only understood by their (now incapacitated) companion, bystanders can misinterpret or ignore the alert, possibly delaying medical attention.

In this dissertation, we develop methods to overcome these three communication barriers.

1.3 Theory of communication

Before attempting to improve canine–human communication it is important to first define what communication is.

Communication can be defined as the act of conveying meaning from one entity to another through the use of mutually understood signs and rules.

We thus begin by recognizing that dogs have shown a well-documented ability to use signs to communicate with each other and with humans [6].

The study of signs and symbols, often referred to as semiotics, identifies three types of classic signs used for communication [7]. We will first define each type of sign (Figure 1) and then show how dogs already use at least two of them for communicating with each other and with humans (Figure 2).

The most basic signs are called *indexical signs* or simply *indices of communication*. These signs derive their name from indices, because like an index, they point to an object. In this case, indexical signs point to the object or entity whose meaning is being conveyed.

The second type, *iconic signs* represent or resemble the meaning being conveyed. For example, a red and orange drawing can be considered as an iconic sign for fire, because both fire and the drawing share color and shape.

Finally *symbolic signs* or *symbols*, and can represent any type of information using abstract patterns that do not necessarily resemble the object of interest.

We can better understand each type of sign by looking at the representation of the concept of *fire* through each type of sign (Figure 1).

In this dissertation, we are interested in the use of the third type of signs, symbols,



Figure 1. The three classic types of signs studied in the field of semiotics.

because we believe their independence of context allows working dogs to communicate important information without relying on sensory inputs the human might not have available (perceptual barriers).

Like humans, dogs can also use signs to communicate information to humans. For example, pet dogs might stand next to a water bowl to ask their human companions for water. Similarly, some working dogs are trained to bite objects known as bringsels to alert about a specific event (Figure 2).



Figure 2. Two types of signs dogs are known to perform. Our focus is strictly on symbolic signs.

1.4 Thesis statement

Having provided definitions of *working dogs*, *barriers to communication*, and *symbols*, we now present our thesis statement:

Wearable computing interfaces can improve communication from working dogs to humans by allowing them to accurately produce symbols in the form of alerts.

CHAPTER 2

LITERATURE SURVEY

We now describe each of the relevant areas of study for developing a canine communication system (Figure 3). The three areas are animal–computer interaction, pattern recognition, and semiotics (the study of signs).

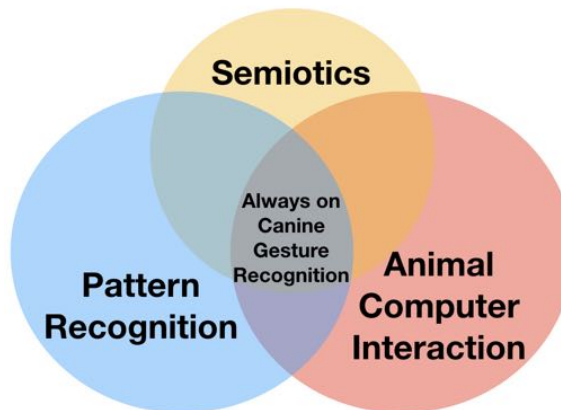


Figure 3. Venn diagram illustrating the intersection of three fields of study relevant to canine–machine interaction.

2.1 Canine communication

Humans have coexisted with dog-like canids at least since the hunter gatherer period more than 11,000 years ago [8]. Since then, communication between humans and dogs has adapted to fulfill a variety of needs. In modern times, working dogs occupy different environments and fulfill needs different than those of hunter-gatherer societies. As these other needs increase in scope and complexity, humans have yearned for alternative ways to communicate with working dogs [5].

2.1.1 Alerting on environmental cues

We previously defined working dogs as canines with one or more specific skill that enable them to perform essential tasks for humans.

Dog–human partnerships often rely on a dog’s ability to perceive the environment with a great level of detail. This perception can be augmented with occupation-specific training to detect a given set of cues. For example, guide dogs can distinguish between a type of obstacle requiring the human to “wait” (e.g., cars) versus a type of obstacle requiring the human to “go-around” (e.g., trashcans) [9]. Similarly, explosive-detection dogs can categorize explosives based on chemical characteristics, most notably between “stable” or “unstable” compounds [10].

2.1.2 Canine–machine interaction

Documented interactions between animals and machines can be traced at least to the experiments of B.F. Skinner in the 1930s. Nonetheless, the possible actions for animals participating in these experiments were often limited by operant conditioning chambers (so called “Skinner boxes”) [11]. Decades later, efforts like the UNAM-CAN project described the first wearable computer adapted specifically for working dogs [12]. This work focused on human–dog communication and it foreshadowed other efforts to overcome distance barriers, like the use of wearable speakers to give remote commands to working dogs [13, 14].

In 2001, Benjamin Resner created an interface to interact with his pet dog from outside the home [15]. This *Rover at Home* project was an early attempt of applying user-centered design principles to animals. This study contrasted with previous experiments where animals, such as pigeons, were only trained to use technology (e.g., levers) to make selections in experiments studying animal learning.

In contrast, Resner’s *Rover at Home* project was focused on creating an interactive system for dog training rather than studying learning or cognition per se. His efforts foreshadowed the modern study of technologically mediated interaction as an end in itself, which nowadays falls under the heading of Animal–Computer Interaction, a term that did not exist in 2001.

2.1.3 Augmented working dog communication

Up to this point, most efforts into augmenting canine–human communication were implemented with pet dogs in a home or laboratory environment.

In 2006, a group of researchers interested in canine-aided search and rescue noticed that “barking does not supply enough information to make decisions” about urban search and rescue scenarios and decided to attach cameras to a working dog harness as part of the *Canine Augmentation Technology* (CAT) project [4]. Even though the cameras did not fulfill the original expectations, this effort marked an early attempt to obtain wearable computer inputs from a working dog, if not canine–human communication in the strictest sense (through the use of signs).

We refer to these methods (i.e., cameras) as *passive input* technology because they do not require dogs to actively alter their behavior for the purposes of communication. More recent passive input techniques include body-worn GPS trackers used during hunting activities. [16]

2.1.4 Symbolic canine–to–human communication

While most efforts studying the production of symbols have focused on non-companion animals (e.g., primates and dolphins), exchanges between dogs and humans have been largely asymmetrical [5]. That is to say, dogs have largely remained as recipients of information, rather than emitters, in human–dog interactions.

Rossi et al. conducted one of the earliest documented efforts aimed at symbolic dog-to-human communication [1]. This “dog at the keyboard” project allowed dogs to make requests by pressing lexigrams on a keyboard (Figure 4). Their results showed that “dogs may be able to learn a conventional system of signs associated to specific objects and activities”. In some countries, search and rescue dogs already use a simple form of symbolic information by biting brightly colored objects called *bringsels* to indicate a find to a human via line of sight [17].

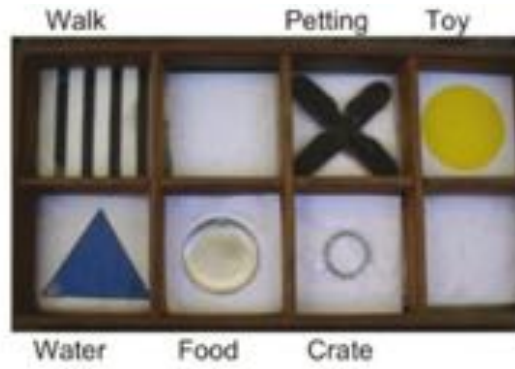


Figure 4. Keys of the lexigram keyboard used by Rossi et al. during their experiments [1].

2.2 Automated sequence recognition

A great portion of the present work falls under the umbrella of sequence recognition. In particular, our second approach to dog-to-human communication (Chapters 4-8) explores automatic detection of gestures sensed from motion on dog collars.

Sequence recognition in general can pose additional challenges compared to problems relying on recognition of temporally static data (e.g., single images). We can illustrate these differences with the classic example of Fisher’s Iris data set. In this well known data set, three types of flowers are measured along four dimensions (the length and width of both the sepals and petals). They are then assigned to one of three classes (Setosa, Virginica and Versicolor). Each flower example is clearly distinguished from others and also distinguished from non-examples. In this case the non-examples do not even form part of the data set. We can say that the Fisher iris data set has no *null class*.

As a result, the data set does not include physical overlap between multiple flowers or any measurements from non-flower objects such as rocks. In contrast, continuous sequence recognition requires grouping multiple measurements and distinguishing them from non-examples (null class). Based on their one-vs-all definition, null classes in sequence recognition tend to contain more measurements than all remaining classes combined. This class imbalance is reflected in the varying amounts of training data for each class and is a problem when applying traditional learning algorithms (e.g., those relying on linear separability) to

sequential data (e.g., time series).

Before examining more current examples of automated recognition we would like to review the history of human–machine interaction in general.

2.2.1 Recognition for human–machine interaction

In the study of human communication, a gesture is defined as “a form of non-verbal or non-vocal communication in which visible bodily actions communicate particular messages, either in place of, or in conjunction with, speech” [18].

The field of automated gesture recognition predates personal computing and can be traced at least to the 1920s with the development of the Theremin, an electronic musical instrument controlled by the body movements of a performer (Figure 5) [19].



Figure 5. The Theremin, an early gesture-based system, is an electronic musical instrument controlled by the movements of a performer.

Since then, motion has remained centrally important to many automated recognition problems. Due to this interest, researchers have developed motion taxonomies to distinguish between types of motion-recognition problems. For example, Bobick et al. divided motion into three categories: *movements*, *activity*, and *actions*. We refer to this classification as the MAC taxonomy.

Over time, the study of automated recognition in computing has allowed an expanded definition of gestures to include any “motion that has special status in a domain or context”

[20]. As we will see, this broader definition does not complement the MAC taxonomy, yet it causes confusion between the notion of gesture and the notion of activity.

Despite these more comprehensive definitions, the notion of communication, whether human–human or human–computer (often called interaction) is still central to gestures and their recognition. Indeed, gesture recognition emerged from the many efforts to minimize the difference(s) between human–human communication and human–computer interaction. This view has been summarized, for example, by Mankoff who stated that “providing support for more natural forms of communication, recognition can make computers more accessible [21]. Such ‘natural’ interfaces are particularly useful in settings where a keyboard and mouse are not available.”

Naturally, the need to avoid the use of keyboards resulted in searching for new methods (or adaptations of existing methods) of providing inputs from continuous signals into computing systems. These methods were eventually known as natural user interfaces.

2.2.1.1 Sequence recognition for user interfaces

Because of the importance of speech in human life, one of the first modes of communication to be adapted for more ‘natural’ human–computer interaction was speech. Later efforts also focused on handwriting as a form of input. Some of these writing systems used simulated pen strokes as their basic elements [22]. Both speech and pen stroke systems could be considered as precursors of gestures aimed at commanding or controlling a system (so-called command and control gestures).

The hands and fingers were then explored independently of handwriting utensils such as pens. The first commercially available hand tracker was the *VPL Data Glove* (1987) by Thomas Zimmerman [23]. These gloves originally relied on optical flex sensors as their main sensing modality, but their creation spurred a great deal of interest in many types of hand gesture systems for interaction. Voyles, for example, proposed hand gesture recognition for human–robot interaction in 1995. He defined a gesture as “an imprecise, context

dependent event that *conveys* the users intentions.” The next year Lee and Yangsheng implemented the first glove system for inertial gesture recognition. It is interesting to note that this paper contains no mentions of the word ‘activity’. As Voyles was describing gesture recognition to interact with robots, the field of computer vision, with its own separate history, saw new applications in sign-language recognition [24]. Up to this point, computer vision practitioners had not been interested in gestures themselves, but in the broader analysis of non-communicative motion, which included general activities [25, 26, 27, 28]. At this point both fields seems to have intersected and by 1997 the effect was clear, a computer vision article describing the taxonomy of movement contained the word gesture sixteen times [29]. Nonetheless, the effect of the interaction between these two fields was not symmetrical. A 1999 literature review of gesture recognition technology, and two subsequent ones in 2007, did not contain a single instance of the word ‘activity’ in its eighty pages [30, 31, 32].

Although the 1997 manuscript states that “human gestures are embedded within communication” it goes on to equate the notions of activities with gestures. This usage gave the impression that gestures were always equivalent to activities, even beyond sign language recognition, and led to the incorrect interpretation of non-communicative movements (e.g., tennis strokes) as gestures [20].

2.2.1.2 *Direct manipulation interfaces*

The next wave of non-vision based recognition systems resulted from the need to control interfaces by mimicking the direct human manipulation of objects, rather than adapting existing forms of oral, written or signed, communication.

In the commercial space this area gained wider interest due to the rise of mobile platforms where keyboard size and computing power were reduced from personal computers and the pointing device (stylus) required a cumbersome two-handed interaction. For example, in this era inertial sensors were used successfully as stand-alone interfaces for palmtop computers. The Itsy system from Compaq used accelerometers to scroll and zoom views on

the screen [33]. This type of efforts ultimately led to new movements being introduced as interactions for track pads (e.g., two-finger scroll), touch screens (e.g. pinch and zoom), remote controls (e.g., Nintendo Wiimote) and vision-based systems (e.g., Microsoft Kinect).

2.2.1.3 Automated gesture recognition for user interfaces

We have previously traced the history of automated recognition systems for speech, stroke, and motion for application-specific interactions. The underlying recognition methods have varied for each system. In surveying the literature, we found two previous efforts that relied on similar approaches to those presented in this dissertation.

One of these is a state-based technique for the representation and recognition of general movements to be assembled into larger groupings [20]. The other is a recognition algorithm that could categorize simple motions to be later combined by an application designer [34]. This last system was inspired by use cases not requiring always-on interaction (e.g., gaming) and even with those there were no documented user evaluation at that time. Nonetheless, the ideas of detecting atomic movements and the notion of states are ones that we rely on heavily in this work.

We now focus on reviewing application systems that created new movements for interaction (e.g., shake to shuffle) as opposed to movements that had a pre-established meanings before the creation of the system (e.g., sign language). The purpose of this survey is not to be exhaustive but to provide examples as to different types of recognition systems along with their benefits and drawbacks.

We begin by considering efforts of Schlomer et al. at creating arbitrary, gesture-like movements using a Nintendo Wiimote controller [35]. To this effect, they defined a series of five movements to be performed by grasping the Wiimote device and moving it in space. These movements corresponded to tracing a series of shapes (e.g., *square*, *circle*, *letter z*, *roll*) or performing a tennis stroke. Processing the accelerometer data using k-means clustering, hidden Markov Models and a naive Bayes classifiers, they performed a recognition evaluation of each movement. The average recognition rate of each movement are

described in Table 1.

Table 1. Reported accuracy of Wiimote movements.

Movements	Reported Accuracy
Square	88.8%
Circle	86.6%
Roll	84.3%
Letter Z	94.3%
Tennis	94.5%

We now move to consider the use of inertially-sensed movements intended for mobile phones [36]. Liu et al. in particular developed a system, *uWave*, for detecting a series of eight movements that were previously identified by users for interaction with home appliances. They relied on a library of 4,480 movement examples from eight (8) participants over multiple weeks. Their recognition evaluation showed that *uWave* achieved a global accuracy of 93.5%

Finally, we note that towards the end of the decade, researchers such as Mankoff explored the use of mediation techniques to handle ambiguous recognition of user inputs in systems. This work described two types of mediation techniques: *choice* and *repetition* [21]. While choice might not be available to working dogs in outdoor environments, repetition is a possible action for dogs to handle incorrect classification, or at the very least, non-detection by the recognizer.

2.2.1.4 Usability of gesture interfaces

We would now like to conclude our survey by addressing some usability concerns raised by human–computer interaction experts regarding gesture-based interfaces and how these concerns might affect our efforts at developing gesture-like interfaces for canines.

In the realm of touch-based devices, experts such as Don Norman, have noted that the onset of ‘natural user interfaces’ has caused designers to disregard previously accepted usability principles [37]. For example, Norman and Nielsen call gestural interfaces a ‘step

backward in usability’. They rate gesture interfaces poorly on metrics of visibility, feedback, consistency, discoverability, reliability, lack of undo, and scaling.

We must note that these metrics carry different levels of importance when considering the end goal of each recognition system. This distinction is particularly true when considering interfaces like the one proposed in Chapter 4, which are not meant to be used for user interfaces, and are not performed on a mobile touch-screen device by a human.

To understand the differences between each metric in regards to our proposed system, we first summarize the types of movements used for gesture-based interaction.

Table 2. Types of movements and their functions.

Type of movements	uses	Example
Continuous-input, continuous output (CICO)	Direct Manipulation Command and control,	Drawing, pinch and zoom
Continuous-input, discrete output (CIDO)	Activation gestures (Always-on) Command and control,	Shake to shuffle, sign language
Discrete input, discrete output (DIDO)	Activation gestures (Always-on)	Arcade Dance Dance revolution (might not even be considered gestures)

Traditionally, movements used in gesture systems have informally been classified as either continuous or discrete. We further refine this definition by subdividing *continuous input continuous output* (CICO) movements from *continuous input discrete output* (CIDO) movements.

User interfaces, especially so called natural user interfaces, typically rely on continuous inputs but discrete outputs to make selections and perform actions. For example, some current touch-based devices use a *swipe left* movement to indicate a rejection of an item on screen. Because there are limitless ways to perform continuous input movement, a large subset of them is not ‘visible’ to the users. For example, how will the user know that *swipe left* is a gesture but not *swipe up*? This ambiguity causes the low scores Norman assigns to gesture interfaces on *visibility* and *discoverability*.

A similar issue arises with the usability metric of *consistency*. This metric measures how often a given action results in a given result across multiple applications. Therefore,

consistency must be enforced by the application designers, and is not necessarily a property of any individual movement itself.

The metrics that most concern the work in this dissertation are *reliability* and *feedback*. After taking their importance into consideration, we have addressed the need for high reliability and feedback in both of the systems we examined (i.e., tangible and gesture interfaces).

In this way, we believe the systems proposed in this dissertation will avoid the pitfalls described by Norman and Nielsen in their study of gesture systems.

2.2.1.5 Automated gesture recognition for always-on interactions

Always-on interaction systems are those that are not designed for one application (e.g., gaming) but are meant to be available to the user at all times. Until the 1990s, always-on interactions had been limited to mimicking existing modes of communication between humans (e.g., speech, writing, sign language, etc.) and not on creating new modes of interaction. For this reason, the literature on always-on recognition is scarce compared to other areas of automated recognition. Because of this imbalance, and despite its many commercial uses, explicit always-on gesture recognition remains a relatively understudied discipline in research settings.

One of the first examples of always-on interaction in its own right was designed in 1993. It came in the form of a device that used acoustic signals to control electrical appliances in the home [38]. This device was later commercialized and known as The Clapper.

With the onset of mobile and wearable computing, the amount of systems implementing some form of always-on interactions increased. A notable example was the ‘shake to shuffle’ gesture of the Apple iPod. This trend continues to the present day, as smart watches have relied heavily on a wrist flicker movement to ‘wake the watch’.

Because the system is always on, the line between gesture and activity becomes blurred, especially because a good gesture recognizer must still accurately reject activities that can cause false positives. In some extreme cases, depending on the definition of a gesture, the

line between the notions of gesture and activity might not exist. This blurring of the lines is especially true in inertial recognition problems. For the purposes of this dissertation, we will sometimes use the word *gesture* to refer to an action that is only performed to interact with a system, as opposed to a daily activity, regardless of whether the action conveys meaning by itself (without the system).

One of the few examples of systems focused exclusively on new movements for gesture interfaces was the work of Ashbrook and Starner et al. [39]. This work presents one of the most, if not the only, complete study of inertial gestures for always-on interaction. It formalizes concepts for gesture discovery, which are typically absent in human-mimicked computer interaction such as speech recognition. The gesture portion of that work relied on a wrist-mounted wearable aimed at finding gestures for so-called *micro-interactions*. These concepts include various notions that will be used heavily in Chapter 4, most notably the minimization of false positives through the use of data sets of everyday movements.

2.2.2 Automated activity recognition

Based on the earlier discussion, it should now be clear that automated activity recognition until the mid 1990s was a separate field from automated gesture recognition. While they increasingly overlap in the present day (especially in the field of ubiquitous computing), we saw that the perception of motion by machines has a rich, and largely separate, history originating in the field of computer vision. We can trace back this research even further to the work of Nagel who began tackling motion understanding problems in 1977 [40].

In summary, activity recognition is interested in the machine perception of motion as an end in itself (e.g., running, walking) rather than as a proxy for communication (e.g., gestures of sign language).

2.2.2.1 Human activity recognition

Before concluding our literature survey, and despite the differences outlined so far, it is pertinent to briefly review the field of human and canine activity recognition in the spirit of

completeness.

As we described earlier, as interest from the different disciplines studying motion continued to increase and overlap, the new field of automated human activity recognition took form in wearable and ubiquitous computing. It incorporated approaches from both vision and non-vision recognition communities. For practitioners not attached to any particular approach, it was the expansion of the activities of interest, especially those performed outdoors, that led to the camera giving way to body-worn (or body-held) sensors measuring inertial motion [41].

One of the early landmark articles on human activity recognition, *Activity recognition from user-annotated acceleration data* was presented by Bao & Intille in 2004 [42]. In it they described a system that used two bi-axial accelerometer to recognize 20 user activities with 84% accuracy.

Two years later, Lester & Choudhury et al. published *A practical approach to recognizing physical activities*. Their focus was on developing the software and hardware elements for an activity recognition system aimed at health-care applications. For eight activities they reported a global accuracy of 90% on twelve subjects. More importantly, they described practical requirements on their systems that would soon become commonplace.

They required the system to have the the following properties:

- Used data only from a single body location
- Should work out of the box across individuals, but should be able to improve with user-specific data
- Should be effective even with a cost-sensitive subset of the sensors and data features.

These requirements remain important in activity recognition today, and heavily influence the present work.

2.2.2.2 *Canine activity (and posture) recognition*

As human activity recognition became more popular, it became increasingly applied to canines and other domestic animals as a method of monitoring. Because activity can be inferred from changes in posture (both in humans and dogs) we consider them together in this section.

One early effort at canine activity recognition detected postures in urban search and rescue (USAR) dogs [43, 44]. The postures were sitting, walking, standing and lying. Here, an offline rule-based algorithm achieved an accuracy of 76%. A subsequent effort attempted to detect postures as part of an automated dog training system to determine whether a reward should be dispensed [45]. This research has since expanded to include changes in postures as well as movement (activities) [46, 47].

Inertial activity recognition has also been used for monitoring the well-being of pet dogs [48, 49, 50, 51, 52, 53]. Commercial products including inertial sensors (among other capabilities) for this purpose number the dozens with new ones coming to market every year. Some examples include the Whistle Activity Monitor, the Voyce Health Monitor, the PetPace Monitor and Heyrex.

2.3 **Comparison and limits of sequence recognition**

Gesture recognition for always-on interaction with inertial sensors has distinctions from other types of sequence recognition problems. The distinctions we described above highlighted differences between the *methods* of recognition. Now we would like to show how these methods are often the result of different recognition problems striving for different *objectives*.

2.3.1 **Objectives of supervised recognition**

The most common objective of supervised recognition is automation. Automation is defined as “the technology by which a process or procedure is performed without human

assistance” [54]. We summarize this objective as ‘removing the human from the recognition loop’. The reason for desiring automation varies from problem to problem, and this reason itself dictates other objectives. It can range from desiring a better understanding of a human recognition process (e.g., facial recognition), to achieving an output faster than a human can provide (e.g., scanning a text document with optical character recognition).

Because of their co-occurrence in supervised recognition, increased *automation* and increased *speed* are often thought of as the same objective. For example, popular instructive resources for newcomers encourage them to tackle “problems a human could solve, but where it would be great if a computer could solve it much more quickly.” In other words “if a human expert could not use the data to solve the problem manually, a computer probably will not be able to [solve it] either.” [55] When stated this way, this view might seem surprising, or contradictory to the spirit of automation, but it is in fact a re-statement of the definition of supervised learning. If the human expert were not able to “solve the problem manually”, then there would be no way to provide the algorithm with annotated training data.

In our construction of a canine gesture recognizer (Chapters 4-8), the goal of automation was not achieving a new understanding of canine motion (although it is perhaps a prerequisite) or an increase in recognition speed (even though it is extremely useful). Instead, our objective was to recognize gesture-like movements with *different* sensor modalities than used by the human.

So far we have listed four closely related objectives of supervised recognition.

1. Minimize the human role in a recognition task.
2. Obtain new understanding of a human recognition task.
3. Increase the speed of a recognition task.
4. Perform a recognition task with non-sensory data (e.g., inertial sensing as opposed to vision).

The fourth objective is common in inertial recognition problems such as activity recognition, where the recognizer tries to estimate the user’s activity without the benefit of *seeing* the human. This fourth objective is sometimes interpreted as performing a task a human expert is unable to. In this scenario, requiring human knowledge and understanding are viewed as problematic for system design.

“Feature extraction for activity recognition [...] is usually a heuristic process, informed by underlying domain knowledge. Relying on such explicit knowledge is problematic when aiming to generalize across different application domains.” [56]

Based on our experience, which is described in the following chapters, underlying domain knowledge will remain present in supervised recognition problems. Even if a given set of features is generated without human domain knowledge, this knowledge will still be necessary later on to evaluate whether the annotation for ground truth labels or classification outputs are correct.

Similarly, even though a domain expert might not be initially necessary, some sensor expertise is certainly required to ensure that sensor data output is correct, and not uncalibrated, disoriented or somehow distorted.

In summary, having correct ground truth annotations requires some combination of human domain knowledge and human sensor knowledge. As a result, the human expert can be removed from the recognition loop, but not necessarily from the design of the system.

2.3.2 Limits to canine gesture recognition

Having considered the importance of ground truth labels, we will see in Chapters 4-7 that there are two scenarios where the human expert might be unable to provide reliable training data to the canine gesture system we developed. First, it will become clear that humans are poor visual inspectors of whether a given gesture movement occurred or not, because dogs can perform the movements faster than the human eye can perceive in the moment,

especially without a video recording of the movement. Secondly, even with the benefit of a video recording, there are no universal gesture definitions like there are, for example, in American Sign Language. This inability posed significant challenges on obtaining ground truth annotations and on training the gesture itself to be performed accurately.

These limitations contrast with computer vision problems where, from the outset, there is general agreement on whether, for example, a car is in a given picture frame or whether a human is walking forward or not. In this case, more annotated data can be obtained from untrained users inspecting and annotating each image. In contrast, we initially lacked a definition of each gesture as concrete as the definitions of ‘car’ or ‘walking’ so these annotations had to be performed by an expert relying on an external reference (video). The inability to outsource the annotation for ground truth labels was an issue of increasing importance.

Secondly, even when inspecting the inertial data itself, rather than visually inspecting the dog’s movements, it is still difficult (even for humans accustomed to examining inertial data) to determine whether a potential gesture occurred at a given time. Untrained humans might not even know the correct time scale to examine and the lack of appropriate annotation tools for small-scale movements can make the problem close to intractable.

Finally, we conclude this section with remarks by Lukowicz et al. on the difficulties of gesture spotting versus isolated motion recognition [57]. “In contrast to isolated motion recognition that has been shown in various areas, the spotting task is much more challenging.”

One of the first difficulties they observed, and we observed it as well, was the issue of co-articulation. Co-articulation is the phenomenon where gestures performed consecutively influence each other. Second, they highlight issues of intra-subject and inter-subject variability, which are exacerbated in dogs, the species with most anatomical variation among its members [58]. Third, the motion events to be spotted may only occur sporadically, in a continuous data stream, while at the same time being embedded into other,

partly arbitrary null-class or partially attempted movements. These movements, however, are “inherently difficult to model, due to their complexity and unpredictability”. As a consequence, conventional recognition schemes, even ones tailored for continuous classification, such as hidden Markov models (HMMs), are “not directly applicable for our recognition task”, since they rely on having appropriate null-class models which are not always present or readily obtained. Consequently, we cannot take advantage of the “implicit data segmentation capabilities that hidden Markov models provide”. Finally, we have to deal with the fact that motion events in the potential gestures we examined are typically very short and every second must be accounted for. This means that for any explicit segmentation-based recognition, exact localization of event boundaries is crucial.

One peculiarity of training dogs to perform movements is that providing descriptive feedback is more challenging when given from human-to-dog versus human-to-human. For example, if a human performs a gesture incorrectly, another human can say ‘that was too slow’, or ‘too wide’. In the case of dogs, it is much harder to provide this type of descriptive feedback, and we rely on discrete (often binary) feedback instead (e.g., beep or click sound).

We will discuss all these issues in greater detail in the chapters that follow. For ease of comparison they have been summarized in Table 3.

In conclusion, throughout this dissertation we will show how creating communication interfaces for working dogs poses new challenges in sequence recognition and wearable computing.

Table 3. Differences between automated recognition problems in related areas.

Type	Peculiarity
Sequences vs single samples	Multiple samples must be grouped as a sequence
Continuous vs isolated	Examples lack boundaries (must window or segment)
Inertial vs image-based	Cannot be annotated directly (require video)
Gestures vs activity	Class occurrence is sporadic and short in duration
Interaction vs monitoring	Must process in an on-board fashion
Online vs offline	Minimal processing time is required for result
Always-on vs application use	Must avoid false positives from other movements
Canines vs humans	User is not part of recognition loop User cannot annotate own data User cannot maintain sensor position Few previous gestures (if any) exist Each gesture must be trained with discrete feedback

CHAPTER 3

WEARABLE TANGIBLE INTERFACES

Similar to the developments of automated recognition systems for human–computer interaction that we examined in Chapter 2, we will begin considering direct manipulation of tangible interfaces as a precursor to gesture-based systems. In this dissertation we refer to body-worn direct manipulation interfaces as *wearable tangible interfaces*.

3.1 Introduction

Working dogs such as those in assistance and law enforcement scenarios use harnesses with equipment related to their role or occupation [5, 59]. For example, guide dogs wear a harnesses with handles that humans can hold as a guiding mechanism. As described earlier, our interviews have revealed that despite the non-trivial communication occurring in dog-human partnerships, the amount of vital information perceived by dogs exceeds their ability to communicate it.

We started by investigating on-body interfaces for dogs in the form of wearable technology integrated into existing harnesses [2]. We created three different types of interfaces that dogs could activate based on common dog behaviors such as biting, tugging, and (snout) touching (Figure 6).



Figure 6. Two participants activating early iterations of each interface [2].

Each sensor in these interfaces was connected to a micro-controller board based on the

ATMega328P microprocessor. The micro-controller recorded readings every loop-cycle to non-volatile external storage (microSD card). Additionally, the micro-controller was wired to a piezoelectric buzzer that produced a beeping sound when the activation condition for each interface was met.

3.2 Resistive bite interfaces

We used force-sensitive resistors (FSRs) and a 3D-printed enclosure to construct three iterations of bite interfaces. The first had a rectangle shape (Figure 7). This shape was covered in a way similar to existing devices known as bringsels, which are used in search and rescue.

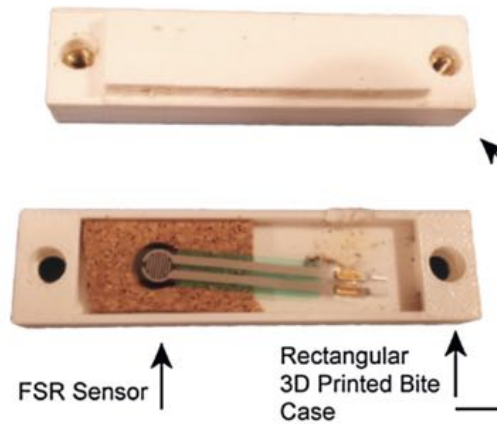


Figure 7. Construction of bite interface from a force sensitive resistor.

The second version had two broader surfaces to induce biting in the necessary vertical direction, rather than sideways. The two surfaces were oval-shaped as can be seen in Figure 8.

The final version had four panels that could be pressured to achieve an activation from any of four directions (Figure 9).

Each bite interface had a 0.16 in (4 mm) diameter active sensing area whose resistance depended on how much pressure was applied. The harder the force, the higher the resulting voltage (Figure 10(b)).



Figure 8. The oval-shaped bite interface improved results by inducing the bite force to be applied vertically.

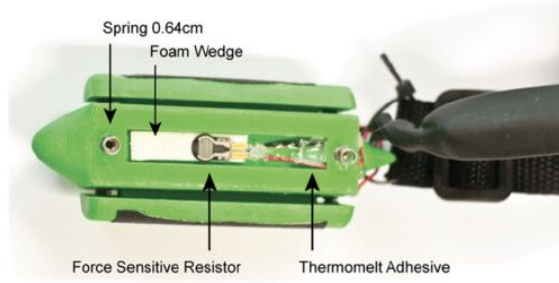


Figure 9. Construction of four-sided bite interface from a force sensitive resistor.

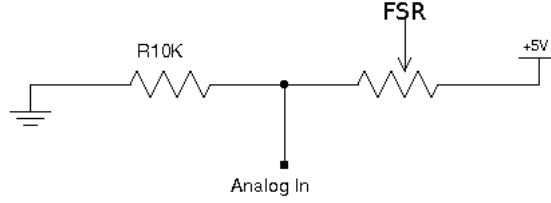
3.2.1 Bite detection mechanism

In the final version, four force-sensitive resistors were wired in parallel ($R_{bite} = R_1 || R_2 || R_3 || R_4$) to a single analog input in the micro-controller. In all previous versions the single force sensing resistor was wired in a voltage divider configuration (Figure 10(a)). When the voltage at $Analog_{in}$ surpassed a threshold of $\Delta T > 750 \text{ units}$ out of 1023, a bite was recorded (and the sound alert was produced).

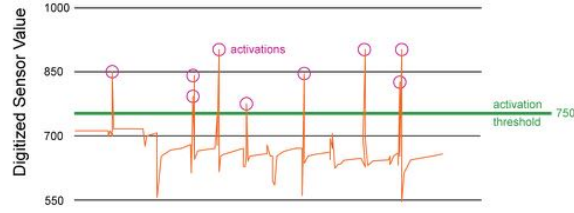
3.3 Resistive tug interfaces

The tug interface consisted of a 10-cm stretchable variable resistor sewn into an elastic band, which was sewn to a braided fleece material as shown in a subsequent section (Figure 11(b)). The dog activated the interface by grasping and tugging the toy with his teeth.

The tug interfaces were designed to be strong enough to compensate for the fragility of



(a) Voltage divider input diagram.



(b) Analog sensor readings over time.

Figure 10. Input configuration for bite and tug interfaces and activation graph for one session with the four-sided bite interface.

the stretch-sensing resistor, yet sensitive enough to register a tug by the dog’s mouth. This compromise was achieved by sewing the resistor into an equal length of elastic. Because the elastic was not as stretchable as the resistor and more durable in withstanding pulling force, it enabled the resistor to stretch enough to change its resistance, but not enough to break it as the dog pulled on it. To activate the tug interface, the dogs reached around and grasped the spherical part of a dog toy, gave a brief tug, and released it upon receiving auditory feedback.

3.3.1 Tug detection mechanism

In early versions of the tug interface, we set an activation threshold at a fixed value for a tug or pull to trigger the corresponding sound alert. Due to the 10-bit analog converter, the stretch values were represented as an integer between 0 to 1023.

Because the sensor could change its angle of orientation during daily use, the baseline resistance, and hence the force required to activate the interface was not constant and thus created confusion among the dogs attempting to activate it.

The final version of the tug interface was contained in an enclosure that allowed intentional re-positioning at different angles and prevented the interface from swinging freely. We called this version the adaptable tug interface. In this final prototype, a single numerical threshold would no longer suffice because the baseline resistance changed with each position. Instead, we needed to analyze changes in the last n samples and set the threshold accordingly. In this case, we empirically determined a change of $thresh_{delta} = 25$ units in the span of $n=10$ samples to be a suitable threshold.

$$tug = [tug_{i-n}, tug_{i-(n-1)}, ..., tug_i] \quad (1)$$

$$\Delta tug = tug_i - tug_{i-n}$$

From this Δtug , we created the activation criteria:

$$Activation = \begin{cases} False & \Delta tug \leq thresh_{delta} \\ True & \Delta tug \geq thresh_{delta} \end{cases} \quad (2)$$

3.4 Touch interfaces

Our touch interfaces used infrared sensors with an analog output set to detect movement at a distance of 3 cm (Figure 11(c)). The dogs were expected to touch their snout directly over the sensor to activate it. The infrared sensors were wired to one of the analog pins on the micro-controller to capture the values of objects moving toward and away from the sensor. In some cases, dogs learned that waving their snout at a certain distance, without touching, was sufficient to obtain trigger the activation criteria. This behavior was a foundation for some of the gestures we explored in Chapter 4.

3.4.1 Touch detection mechanism

To detect the distance of objects from the sensor, the micro-controller implemented a moving average of $n=50$ readings and produced a beeping sound if that average was lower or

equal to the preset threshold. The buzzer would beep if an object was in front of the sensor for half of a second and turned off once the object moved away approximately 18 cm.

$$\frac{1}{T} \sum_{n=1}^T proximity_i \quad (3)$$

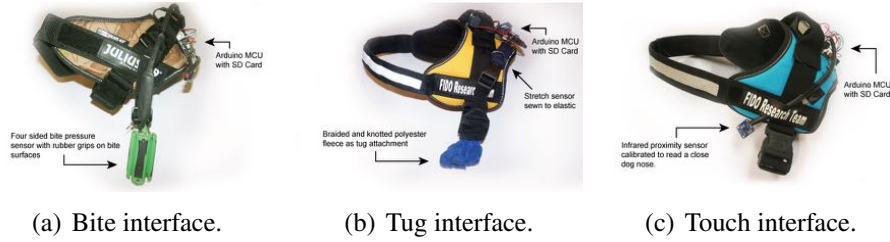


Figure 11. Detailed illustration of early versions of the instrumented harnesses and their components.

3.5 Protocol and participants for wearable tangible interfaces

We tested these interfaces with eight dogs previously trained for a variety of occupations and compared their effectiveness in several dimensions [2]. The skills and occupations of the dogs we recruited were not always identical to our target occupations (guide and search and rescue). This difference is based on our recruitment process which emphasized the need for participant availability and preserving the integrity of existing training in active working dogs. We selected participants based on the following criteria:

- Familiarity with behavior-reward training scenarios.
- Availability of the dogs and their human companion.
- Proximity to the testing location in the greater Atlanta area.
- Ability to participate without compromising previous training.

We can see the ‘demographics’ for each subject below (Table 4)

3.6 Method

We relied on two types of experiments to test metrics pertaining to *true positives* and *false positives*. The true positive experiment consisted of the human trainer sequentially asking

Table 4. Participant demographics for the tangible interface study.

Dog	Breed	Training	Age	Weight (lbs)	Weight (kg)
BC1	Border collie	Assistance, agility, wearables	5	47 lbs	21.3 kg)
BC2	Border collie	Assistance, agility, wearables	4	33 lbs	15 kg
BC3	Border collie	Agility	7	40 lbs	18 kg
BC4	Border collie	Agility	16	33 lbs	15 kg
BC5	Border collie	Agility	3	32 lbs	14.5 kg
R1	Retriever	Agility, wearables (limited)	5	72 lbs	32.6 kg
R2	Retriever	Agility, wearables	5	70 lbs	31.7 kg
PB1	Pit bull	Agility	5	49 lbs	22.2 kg

the dog to activate the interface approximately ten (10) times. After each correct interaction, the human would mark it as such using a ‘click’ sound generated by a mechanical device known as a ‘clicker’.

Each dog participated in at least one training and one testing session for each interface. All training and test sessions were video recorded for later analysis. Training sessions began with the interface being off-body until the dog was comfortable with the interaction required. When the dog was proficient interacting off-body, we put the instrumented harness on the dog and trained him to find and activate each interface on his left rib-cage area. When the dog was consistently operating the interface on-body, we provided a period of rest before moving on to the testing session. Both training and testing sessions were less than five (5) minutes, some considerably shorter.

For the false positive experiments, each interface was worn by a dog as they walked on-leash through an outdoors environment for a span of thirty minutes. They wore harnesses with each interface while performing normal working dog actions such as walking outside on a hilly, forested path. We recorded both the dogs (video) and the interface (sensor values) for the entire thirty minute period. The dogs were allowed to perform normal behaviors, such as shaking and sniffing. The dogs were not asked to deliberately activate the interfaces during the false-positive testing.

3.7 Performance metrics

During the analysis phase, we found it necessary to define specific types of accuracy to account for unforeseen cases. These cases included the dog performing the incorrect action on a given cue, activating the interface more than once per cue, or unsuccessfully trying to reach the interface.

Cue Response Accuracy: describes how well the dog responds to a cue to interact with an interface.

Interface Detection Accuracy: describes how well the system was able to detect a correctly performed activation from a given interface.

Interface Reachability: describes how well a dog was able to reach or access a given interface. Because this metric can affect all others, we examined it in greater detail in a follow-up study [60].

Table 5. Definition of terms for each of the three performance metrics.

	Total (N)	Deletion	Substitution	Insertion
Cue Response Accuracy	Cues	Dog ignored cue	Dog performed incorrectly	Unrequested dog action
Interface Detection Accuracy	Interactions	False negative	Incorrect detection	False positive
Interface Reachability	Reach attempts	Unsuccessful reach		

3.8 Results

We summarize Interface Detection Accuracy for each version of the interfaces in Table 6. During the outdoors experiment, only the rectangle bite interface and the touch interface had instances of false positives [2].

3.9 Discussion

In this study we demonstrated that it is possible to create wearable tangible interfaces that dogs can reliably activate on cue, and determined physical factors that affect dog success with bodyworn interaction technology. We observed that dogs had more understanding of the task than we anticipated. For example, they would hold and interact with an interface

Table 6. Interface Detection Accuracy for each type of interface.

Subjects	Rectangle Bite	Oval Bite	Four-sided Bite	Infrared Touch	Tug Fixed	Tug Suspended	Tug Adaptable
BC1	64%	88%	75%	92%	64%	73%	92%
BC2	64%	77%	91%	77%	100%	42%	91%
BC3			92%	100%		95%	
BC4			67%	89%		67%	
BC5			89%	100%		64%	
R1	0%	50%	64%	100%	100%	0%	91%
R2			67%	100%		38%	
PB1			92%	100%		100%	
Avg	44%	78%	80%	92%	85%	60%	92%
FP/hr	4	0	0	10	1	2	0

multiple times to ensure a correct activation, particularly if the previous activation had gone undetected.

For demonstration purposes, we also trained one dog to discriminate between two distinct stimuli that a hearing dog might perceive before activating one of two interfaces. A hearing dog is a dog that assists humans with hearing impairment by taking them to the source of a given sound. Because the source of some sounds can be dangerous, we successfully trained a dog (BC1) to selectively activate one of two interfaces depending on the sound cue. A doorbell ring required activating the tug interface while a fire alarm required activating the touch-based interface. Because of the effort required to reach the interfaces in the rib cage area, we did not train activations that required multiple reaches in one activation. As such, we did not obtain more than one bit of information per interface.

3.9.1 User feedback

We conducted a series of personal interviews with humans who partnered with working dogs. These interviews included two sessions with two explosive detection practitioners, two sessions with one search and rescue specialist, and two sessions with two assistance dog users. The questions asked pertained to the prototype as well as reflections on their practice.

These interviews revealed excitement over the possibility of unambiguous dog-to-human

communication, but expressed concern about the practicalities of wearable tangible interfaces. For example, a user with an assistance dog expressed concern that her dog might not be wearing his harness inside the house. A search and rescue trainer expressed that her dogs do not wear harnesses during a search because it might limit their mobility. Finally, our interview with an explosive-detection squad revealed that their dogs already have substantial equipment and are unlikely to benefit from adding additional weight, or from the risk that the interfaces get caught in the environment.

We also observed that the sensing components that were acted upon (force and stretch sensing resistors) degraded over time and required compensating for this degradation with more intensity in the interaction as time went on. This degradation was frustrating to the human and required substantial effort from the dogs. To address these concerns, we decided to explore an alternative approach to interaction that relied instead on recognition of movements with inertial sensors. This second approach involves wearable gesture interfaces.

CHAPTER 4

WEARABLE GESTURE INTERFACES

Having considered direct manipulation in wearable tangible interfaces, we can now mimic some of the movements used in these interactions to create gesture-like movements for wearable gesture interfaces.

4.1 Introduction

In contrast to our first approach of creating a dedicated communication channel to reduce communication barriers between working dogs and humans, this second approach aims to re-purpose parts of an existing channel (motion) to generate the alerts for communication (Figure 12).

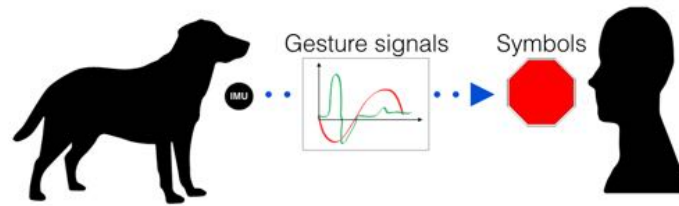


Figure 12. System diagram of a gesture-based communication system.

As we briefly mentioned in the previous chapter, gesture systems can greatly reduce the size of the wearable system, from a fairly large harness to a comparatively small device on the collar. This form factor could be better suited for situations where a harness might be too cumbersome, hot, or dangerous to wear.

Additionally, the selection space for interactions in gesture-based systems is potentially much larger than interactions for direct manipulation interfaces, because the latter are limited by the available space on a harness and the ability of the dog to reach the interface.

4.2 Harness-based offline system

As we transitioned from tangible interfaces to gesture interfaces we began considering instrumenting the harness with motion sensing equipment. The first method we employed was to collect inertial data from dogs using the on-body Axivity sensor platform developed at Newcastle University [3]. This device includes a three-axis accelerometer (but not a gyroscope in this version). It was attached to the front of a Julius K9 harness (Figure 13). The placement of the sensor on the neck (as a collar) was postponed due to the uncertainty of the effect of the sliding collar motion on the readings.



Figure 13. We used the Axivity accelerometer (circled) in this first experiment.

With the accelerometer attached to the harness, each dog was instructed to perform certain movements using behavior-reward scenarios. Multiple repetitions of these movements were video-recorded, synchronized in time, and annotated using the ELAN annotation toolkit. These sessions of raw data were annotated with the labels corresponding to the certain movements of interest.

Because the set of gestures were not defined at this time, we began by considering a broad range of movements as a baseline for recognition. Originally, the study intended to explore four movements that should be readily observable from inertial data. The movements were *spin* (360° rotation clockwise), *twirl* (360° rotation counterclockwise), *roll-over* and *jump*. Even though these movements are performed without previous training, and are

not gesture themselves, they helped us better understand everyday canine movements.

As we annotated the video sessions, we noticed distinct patterns in acceleration data corresponding to other changes in postures and decided to label these as well for completeness. These extra annotations corresponded to *sit-from-stand*, *stand-from-down*, *sit-from-down*, *down-from-stand*, *down from sit*.

In other cases, we noticed different versions of an activity that we previously thought as the same, and decided to label them separately. The most notable example of this was *roll-over*. Originally we conceived *roll-over* as a single movement. It turned out that some dogs only rolled to one side. We also found out that some dogs did not complete the rolling motion, but instead rolled on their back from one side and returned to their original position from the the same side. We called this last behavior *half-roll*. The net result yielded four movements from what was previously considered a single one. These were *half roll+* (to the right), *half roll-* (to the left), *full roll+* (to the right), and *full roll-* (to the left).

We began data collection with two participants, a retriever and a border collie. Their ages were seven (7) years at the time of the experiment. The activities were sampled at a rate of 100 Hz with a range of $\pm 8g$. The data was segmented using a window size corresponding to 100 samples (1 second) and 50% overlap. The one-hundred readings for each accelerometer dimension (ax , ay , az) were concatenated along a single dimension (1x300) to assemble a feature vector. No feature selection techniques were performed at this stage.

4.2.1 Performance and evaluation

At this stage we relied on the Waikato Environment for Knowledge Analysis (Weka) toolkit to perform a preliminary analysis of different classification methods. WEKA is a freely available Java-based machine learning tool. For evaluation of classification methods, we first used a ten-fold cross validation method in continuous streams of data without the null class. It eventually became apparent that using overlapping windows with cross validation over-estimated performance metrics [61].

Classification by random forests yielded the highest accuracy across all techniques for within-subject training and testing (92% to 98% window-level accuracy for nine movements). When the null class was included recognition rates decreased to 62%. This was our first realization that the null class imbalance would be a critical hurdle to overcome in subsequent attempts.

4.2.2 Lessons learned

Because we attached the accelerometer sensor to the harness, the movements that were detectable corresponded to common everyday movements (e.g., lie down, run, jump) and changes in postures (e.g., sit from stand).

After this study, we decided to focus on movements detected from a motion sensor on a dog collar, rather than the harness. Despite our concerns about orientation, we chose the collar placement because canine communication gestures frequently include head movements and they represent little additional overhead in terms of equipment worn by the dog [6]. In contrast, sensing from the dog harness would not capture a great portion of head movements. Instead, sensing from the collar could capture both head and (indirectly) body movements. Finally, consideration of the harness versus a collar was important because we observed a great degree of variation between service dog harnesses depending on their organization. Furthermore, police dogs already have heavy harnesses which would make it difficult to add more weight, and search and rescue dogs often wear no harnesses at all.

This study was also our first encounter with some of the key differences between automated activity recognition and gesture recognition, as stated in the literature survey of this dissertation (Chapter 2). We review these differences once more in the interest of completeness.

1. Users are unable to annotate their own data.
2. Users are unable to re-position the sensor if dislodged.
3. Movements are non-periodic and short in duration.

4. Dogs are not expected to modify their behavior to increase precision and recall as much as humans (although they perform more attempts if given feedback).
5. The gestures must be communicated non-verbally through training using discrete feedback.
6. No universal gesture definitions exist.

4.3 On-board collar-based system

For our next study we developed an on-board system with simpler classification techniques. The idea was to replicate the threshold mechanism of our tangible interfaces as a stepping stone for addressing the null class challenges presented in our first (offline) system.

4.3.1 Participants

For this second pilot study, we recruited three dogs previously trained in allergy alert, assistance, and police work. The demographics of the participants can be observed from Table 7 below.

	S1	S2	S3
Breed	Retriever cross	Border collie	Belgian Malinois
Training	Assistance	Allergy alert	Explosive-detection
Sex	M	M	M
Age (yrs)	0.5	5	4
Weight (kg)	21.0	21.3	22.23

Table 7. Subject demographics. *Retriever cross* denotes a cross between labrador retriever and golden retriever.

The skills and occupations of the dogs we recruited were not identical to our target occupations (guide and search and rescue). This difference is based on our recruitment process which emphasized participant availability and maintaining the integrity of existing training in active working dogs. We selected participants based on:

- Availability of the dogs and their human companion.
- Proximity to the testing location in the greater Atlanta area
- Ability to participate without compromising previous training

4.3.2 System and equipment

The main piece of equipment used for this study was a commercially available inertial sensor, the WAX9, by Axivity Inc [62]. Contrary to the AX3 in the first study, this unit consists of a 9-axis sensor, including three axes of accelerometer, gyroscope and magnetometer. Only the accelerometer and gyroscope (on the collar) were used for data collection during this study. The WAX9 has the ability to stream data wirelessly, and was set to record at a sampling rate of 5 Hz.

We selected the WAX9 due to its light weight compared to sensors with similar capabilities. Considerations of weight were extremely important because a heavy object might obstruct the intended movements or cause general discomfort during everyday use.

The unit was strapped with two rubber bands (Figure 14) and padded with polyurethane foam to avoid any movement relative to the collar. The position and orientation remained consistent for all subjects. For stability, the collar with the sensor was placed above each dog's existing flat collar.

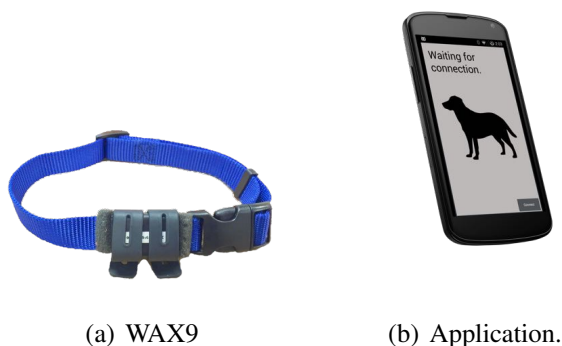


Figure 14. WAX9 inertial sensor attached to a dog collar by two rubber bands supported by an equally sized polyurethane foam piece.

We developed a companion mobile application on a smart-phone device (i.e. Nexus 4 smart phone) running the Android 4.4.4 operating system throughout this experiment. The application received sensor readings at 5 Hz via a Bluetooth connection and played synthesized audio messages corresponding to the gesture movement being performed. The

messages were voiced by the Android Text-To-Speech (TTS) engine. If the device went out of range, for more one second (five samples skipped), a corresponding message saying “device out of range” was communicated to the user. This mechanism was necessary to distinguish between lack of connection and lack of detecting a gesture movement. The smart-phone application displayed a silhouette image corresponding to the basic dog movements detected as a supplemental resource for humans participating in this experiment.

4.3.3 Movements of interest

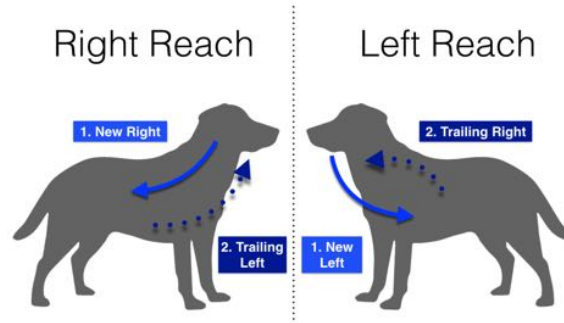
Once we settled on the collar placement we considered the potential neck or body movements we could use to form gestures. Overall, we relied on movements dogs could perform in-situ as opposed to movements that required displacement forward or backward. Candidate gestures involved horizontal movements of the head (“horizontal movements”) and vertical movements of the head (“vertical movements”). The most basic of these involved moving the neck left, right, up or down.

We also considered movements of the body along these dimensions as in the first study. For example, *roll* (as in flight dynamics or “roll over”) and *spin* or *twirl* (360° rotations in either direction of the yaw axis). We called these “rotational movements” (Figure 15). Note that, a similar rotational movement on the x axis (pitch) would amount to a back flip, which we did not consider.

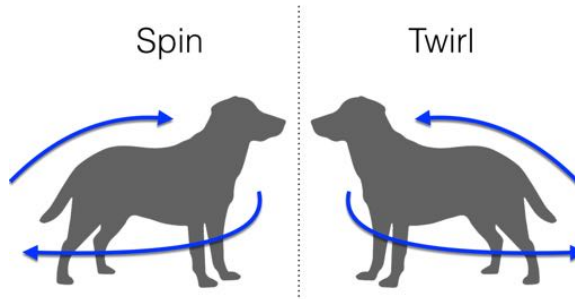
Although our preliminary results showed that *rolls* could be identified by detecting the inversion of the sign of the gravity vector constructed from accelerometer readings, even well-trained dogs seemed uncomfortable rolling on the bare floor and it was not clear that working dogs wearing a leash or harness would be able to roll on a given surface. For this reason, these movements were not considered further as candidate gestures.

4.3.4 Gesture training protocol

The present study depended on training each subject to perform a set gesture movements as described above.

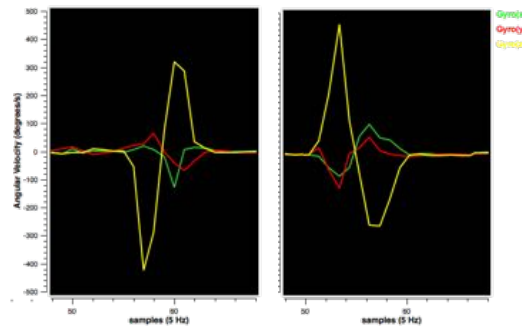


(a) Horizontal movements.



(b) Rotational movements.

Figure 15. The companion application has a visual representation of the basic movements for the purposes of training. Note that spin and twirl are not purely head-only gestures but also involve body movement.



(a) Left reach.

(b) Right reach.

Figure 16. Gyroscope measurements for left and right reach with a completely planar movement.

4.3.4.1 Identification of horizontal movements

Basic horizontal movements consisted of turning the head left or right. Over time we refined this movement to consist of moving the head in a way aimed at touching the nose to the middle part of the ribcage area (similar to the tangible interface interaction). If

the head is moving in a perfect horizontal fashion, this movement will register in the z axis (yaw) of the gyroscope. When this value ranged between predetermined thresholds ($gz < -140$ degrees per second), a *left* gesture is detected. As the head is returning to its forward-position ($gz > 140$ degrees per second), a *right* gesture is detected. We noticed that sometimes the dogs head would not return to their original forward-looking position. So, we refined the gesture definition such that a combination of *left* and *right* movements in close succession resulted in a *left reach* gesture. Similarly, a pair of basic left and right movements resulted in a *right reach*. As described above, we determined these thresholds by observing the measured movements of three (3) dogs.

Once a movement was started, there was a *time-to-live* for each movement to be considered a gesture movement. If this time expired, no compound or gesture sequence would be detected. For this experiment, the *time-to-live window* for reaching to the side was set to $w=3$ seconds

This scheme assumed no lateral preference (the canine equivalent of human ‘handedness’) on the part of the dogs. Although this assumption did not always hold true, we used the same movements and intensity thresholds on each side to avoid excessively tailoring to our subjects.

4.3.4.2 *Excluded basic movement types*

Other types of movements, on which we ultimately did no further experimentation, were vertical movements (e.g., looking up or down). Unlike horizontal movements, where the motion is short in duration and is rarely sustained, vertical movements can be sustained indefinitely. As a result, detecting these movements required system discrimination between the static *posture* of looking up versus the *movement* of looking up, because only the latter would qualify as a true movement. To this effect, we tried to restrict the unit of analysis of all candidate gestures to consists of transitions between one posture to another.

These strategies ultimately proved unsuccessful for several reasons. First, vertical movements proved too difficult to train and perform reliably. In addition, they had a strong

propensity for false positives during everyday behaviors (looking down more so than looking up). The readings also seemed vary by dog size much more so than horizontal movements. Finally, there was substantial overlap between horizontal and vertical movements because they are not mutually exclusive. For example, when performing a horizontal movement, the dog's head will rarely move along a perfectly horizontal plane. Although there are ways to account for this problem, such as redefining the gestures with constraints on all three planes, we postponed such efforts until first testing the simplest set of movement definitions.

4.3.4.3 *Identification of rotational movements*

As in the first study we also considered rotational movements as potential gestures. These rotational movements consisted of *spin* and *twirl*. These are 360 degree rotation to the right and left, respectively. These were inertially detected when a rightward ($gz < -90$ degrees per second) or leftward ($gz > 90$ degrees per second) motion was detected for a sustained period of time. Each movement was monitored by a variable that expired every second unless a subsequent movement was detected. At the point where one second of rightward movement had elapsed (five rightward samples detected at 5 Hz), a *spin* was recognized. Unlike *left reach* or *right reach*, these movements do not occur in left-right pairs. Once again, we note that rotational movements also consists of movement of the body in addition to the head. These heuristics are visually summarized in Figure 17.

4.3.4.4 *Provisional solutions*

During our experiment preparations, we had to provisionally accommodate for two issues not foreseen in the initial design. We describe these solutions for completeness, but acknowledge their limited generalizability for other scenarios.

First, if the movements exceeded an acceleration threshold of $2g$ in as measured by the accelerometer, it was likely that the dog was not in position to perform a gesture, but more likely performing an activity closer to running. At this point the companion application

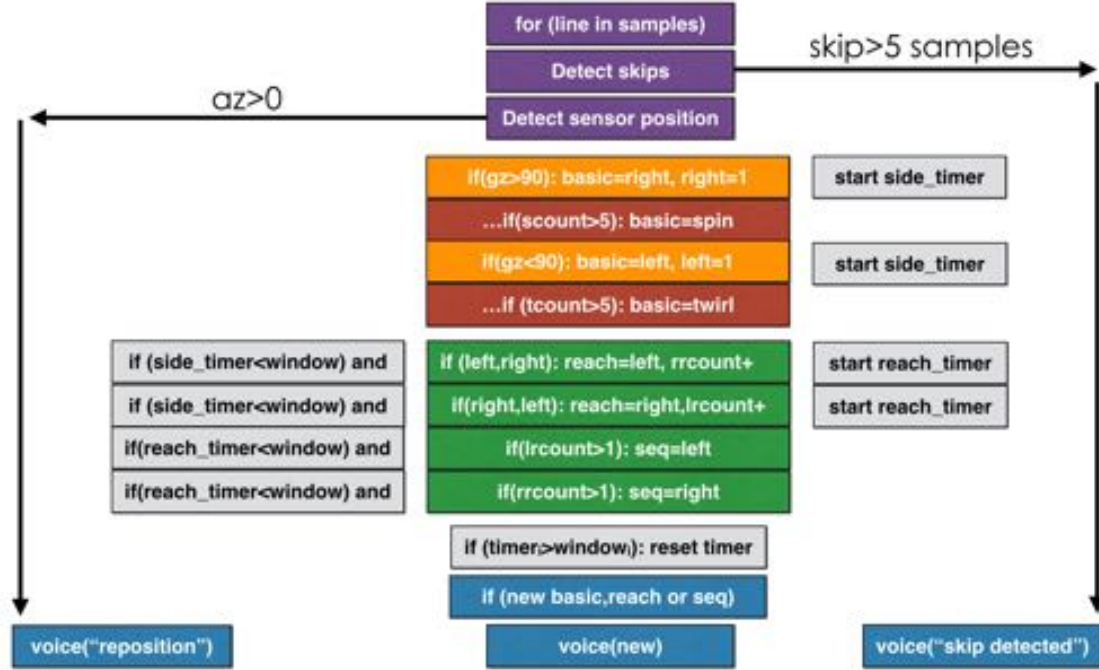


Figure 17. Basic scheme to detect gestures with gz representing the yaw axis of the gyroscope.

simply voiced “too fast” and slept for one-second rather than making an irrelevant or incorrect prediction. We determined this threshold provisionally by observing readings over two one-hour sessions with two different dogs engaging in high and low intensity activities.

Secondly, if the accelerometer readings suggested the collar had shifted from its original position, the system voiced a “reposition” command and slept for one second. Misplacement was judged by the z axis of the accelerometer being greater than zero. We discuss an alternative to address this misplacement in the following study (Chapter 5).

4.3.4.5 *Movements selected*

Based on these experiments, we identified a series of movements that deserved further experimentation (Table 8). To differentiate them from everyday movements, we added a repetition component to the horizontal movements to arrive at this list. We had earlier described the use of repetition to increase reliability as presented by Mankoff [21] and Ashbrook [39].

The final list of candidate gestures for this second study was: reaching twice to the left

side, reaching twice to right side, rotating clockwise (*spin*) and rotating counter-clockwise (*twirl*). One of the constraints we noticed, and will discuss further, was the difficulty of dogs remembering a particular gesture movement. To take this constraint into account, we established a criteria of no more than two repetitions per movement sequence. Regardless of whether dogs are able to count repetitions, we only assumed they can be trained to perform two repetitions in a sequence until further acknowledgement is provided.

Table 8. Summary of all the movements analyzed in this study.

Movement	Type	Description
Right	Basic movement	Rightward movement
Left	Basic movement	Leftward movement
Spin	Basic gesture	Sustained rightward movements
Twirl	Basic gesture	Sustained leftward movements
Right reach	Basic gesture	Reaches to the right ribcage and back
Left reach	Basic gesture	Reaches to the left ribcage and back
Double right reach sequence	Compound gesture	Two right reaches in sequence
Double left reach sequence	Compound gesture	Two left reaches in sequence

4.3.4.6 Identification of gesture sequences

When basic movements are consecutively performed within the span of a certain amount of seconds (*window*), a gesture sequence (also known as compound gesture) is detected (Figure 18). One example of a movement sequence is a *double left-reach*. Basic movements that are not part of the sequence must not be performed while a sequence is in progress, otherwise the counters for compound gesture movement detection will reset. Note that the basic movements (e.g., basic left or basic right) do not necessarily have to be candidate gestures themselves. Both *spin* and *twirl* were composed of movements to the right or left that no longer met the definition for *right reach* or *left reach* gesture movements.

4.3.5 Experimental procedure

To evaluate the proposed movements and the second prototype of our system, we placed the dog collar with a WAX9 sensor above the manubrium on each one of our participants. The sampling frequency for all measurements was 5 Hz at a range of +/- 2000 dps (degrees

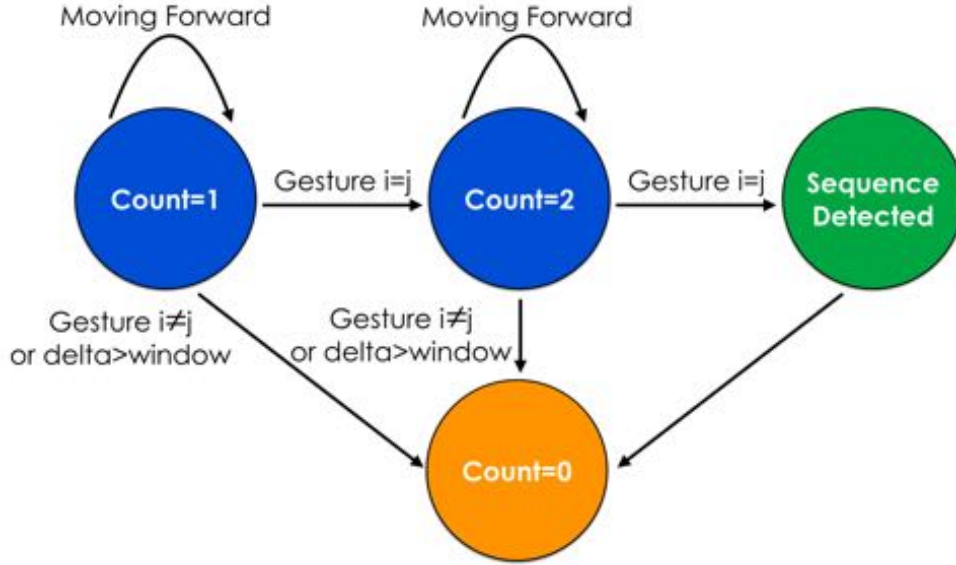


Figure 18. Finite state machine for detecting gesture-like sequences. Dog trainers refer to these sequences as behavior chains. Gesture movement i and gesture movement j represent names of two consecutive gestures

per second) and $\pm 8g$. Each of our participants was subsequently asked to perform at least six repetitions of each of the movements by their trainer. After completing each movement, the experimenter provided a food or play reward.

4.3.5.1 Performance metrics

As in our tangible interfaces, we used separate metrics for *System Accuracy* and the *Cue Response Accuracy* (ease of guiding the dogs to perform the correct movement). Beyond quantifying ease, it is necessary to compute such a metric to accompany the system performance because any failed detection could be attributed to the dog not performing the movement ‘correctly’. For example, it would be unfair to penalize the dog and trainer for not performing a movement on a perfect horizontal plane or at a speed of one (1) degree per second less than a required threshold. For this reason, we show how the human annotator scored both System Accuracy and Cue Response Accuracy to give the reader an idea of the breakdown.

As stated above, Cue Response Accuracy represents how well the trainers were able to guide the dog to perform the desired or correct movement. For Cue Response Accuracy,

we associated penalties with ignoring cues, performing the wrong cue or performing a candidate gesture spontaneously. These three items correspond to our deletions, substitutions, insertions, respectively. This type of metric is used in speech recognition and is known as word accuracy. Similarly, for system accuracy, penalties were given for undetected movements, incorrectly detected movements, or false positives (deletions, substitutions, insertions). All sessions were video recorded and analyzed at a later time to compute their performance metrics (Table 3).

Table 9. Definitions for performance metrics for the second study.

	Cue Response Accuracy	System Accuracy
Total N	cues given	gestures performed
Deletions	cues ignored	gestures undetected
Substitutions	incorrect activities performed	incorrect detections
Insertions	spontaneous activities	false positives

We computed the system’s accuracy from the recorded video based on the previous definitions.

$$WordAccuracy = \frac{N - Substitutions - Insertions - Deletions}{N} \quad (4)$$

4.3.5.2 *System accuracy metrics for sequences*

We now move to the scoring of gesture sequences. This scoring is more complex than earlier metrics because it depends on the underlying basic movements and compound gestures being both detected correctly. To illustrate this, each result table contains two columns for tabulating accuracies of the sequence detection. If a basic movement was undetected, that deletion would also affect the sequences accuracy (Table 10, Table 11).

To account for this case, we computed a second metric (sequences II) where no penalty was given to the sequences for deletions at the basic level.

Table 10. Sequences depended on the correct detection of single gestures and compound gestures.

Stage	Time1	Time2
Dog performed	right,left	right,left
Basic detection	right,left	right,left
Compound detection	right reach	right reach
Sequence detection	right sequence	

Table 11. Results for a new metric, sequences II that only counts cases where the basic units were detected correctly.

Stage	Time1	Time2
Dog performed	right,left	right,left
Basic detection	right,left	right, none
Compound detection	right reach	none detected
Sequence detection	none detected	

4.3.5.3 Repetition experiment

For this study, the dog handler used a *target stick*, *target toy*, or a food target consisting of a small treat, to give the subjects an indication of how to move their heads to perform each movement. Target sticks are commercially available and are in common use in agility practice and obedience dog training practice. Although the resulting motions for each target device exhibit some variation, they were considered equivalent for the purpose of this experiment. If the target involves a food reward, then this method of eliciting a behavior is referred to as *luring*. Luring is a training technique where the human guides a dog to perform a certain action by luring them to follow a food reward.

4.3.5.4 False positives in urban environment

We finally tested the system in a more realistic scenario inspired by active service dogs. In particular, we focused on assistance dogs (guide dogs included) who must accompany their humans as they travel through dense urban environments. Although the leash prevents the testing of gesture detection, this scenario allows for testing of false positives (Figure 19).



Figure 19. Assistance dog walks through sidewalk while wearing the instrumented collar.

4.3.5.5 False positives in open environment

For this experiment, we allowed dogs to run off-leash in an open environment. During this time, they were given objects to fetch and retrieve by their trainer.



Figure 20. Explosive detection canine performing the open environment experiment.

4.3.6 Results

We now present the results of each one of the experiments in this study.

4.3.6.1 Repetition experiment

Our evaluation of the accuracy results for the repetition experiment is summarized below. These were analyzed and computed by a single observer and were subsequently verified by a secondary observer.

Table 12 shows the ease of guiding the desired gestures in each dog. The most common

Table 12. Cue Response Accuracy for each subject. Note that the training methods used were different for each one.

Term	S1	S2	S3
Method	Target stick	Luring	Target toy
Total N	47	48	23
Deletions	0	0	1
Substitutions	8	1	0
Insertions	4	4	1
Accuracy	74%	89%	91%

substitution was performing a full rotation when a side reach gesture was being induced. The second most common substitution, particularly for S1, was reaching to the opposite side of the target stick. When performing a sequence, S1 would wait to be rewarded for the first activity and not perform the second repetition. All of the insertions for S1 consisted of attempting to reach the target stick before it was placed on its intended location.

In cases where S1 performed an inserted or substituted a gesture, we still evaluated the system's detection. In some cases, the unintended gesture movement affected the timing calculation of the more complex gestures and this was scored accordingly (Table 13). For this reason, basic movements (left, right) had the highest accuracy compared to gesture movements (*left reach*, *right reach*, *spin* and *twirl*) or sequence detection (*double left reach* or *double right reach*).

Table 13. System accuracy for S1. Sequences II analyzes the detection of gestures sequences by controlling for cases where the compound gesture should have been detected.

Gestures	Basic	Compound	Sequences I	Sequences II
Total N	82	35	11	11
Deletions	0	2	0	0
Substitutions	7	2	0	0
Insertions	0	3	2	2
Accuracy	91%	80%	82%	82%

Most of the system deletions observed with S2 (Table 14) can be accounted by two factors. First, the *left reach* and *right reach* gestures were performed using the technique known as *cookie stretches*, a form of luring. As we described earlier, luring is a training

technique where the human guides a dog to perform a certain action by luring them to follow a food reward. In the case of S2, he was guided to perform the movements by following the path of the treat reward.

When S2 performed the reach, the right motion was detected appropriately. After doing so, he would look downwards to ensure that no part of the target treat was on the floor (Figure 21). At this point, the time-to-live for the initial reach gesture would expire. When the dog finally came back to face the human, this motion was treated as a new basic movement, rather than the closing part of an existing one. Not only did this phenomena cause the performed reach to go undetected, but it also caused the subsequent one to be interpreted as the opposite side. For subject S2, this error resulted in 16 system substitutions (Table 14). This behavior also explains the large discrepancy between detection of basic movements and detection of compound ones.

Table 14. System accuracy for S2. Sequences II analyzes the detection of gestures sequences by controlling for cases where the compound gesture should have been detected.

Gestures	Basic	Compound	Sequences I	Sequences II
Total N	47	36	14	5
Deletions	0	1	9	0
Substitutions	5	16	0	0
Insertions	1	3	2	0
Accuracy	87%	44%	36%	100%

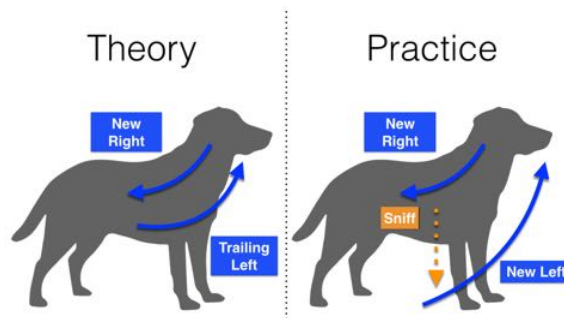


Figure 21. The use of a treat to lure the reach gestures caused problems with detection.

We also noted that dogs trained in occupations requiring constant eye contact with a human maintained this eye contact as much as physically possible while performing the

movement. This behavior was not foreseen in the design of our detection method or in the design of our experiment. If the head is vertically oriented (e.g., looking at the trainer) the movement will be increasingly reflected in the pitch axis of the gyroscope rather than yaw. Nonetheless, horizontal movements are rarely aligned perfectly along a single axis. To account for this issue, we subsequently combined the z axis readings of the gyroscope with the x axis readings, by taking the Euclidean distance between each of the two points and keeping the sign of the z axis. Below are the results of this modification on testing with S3 (Table 15).

Table 15. System accuracy for S3. In this case, gz and gx were combined.

Gestures	Basic	Compound	Sequences I	Sequences II
Total N	49	20	5	5
Deletions	0	2	0	0
Substitutions	0	2	0	0
Insertions	0	3	1	1
Accuracy	100%	85%	80%	80%

4.3.6.2 *False positive experiment*

We performed the false positive experiment with three dogs (S1, S2, S3), under different conditions under which no gesture should activate. The results of these experiments are summarized below (Table 16).

4.3.7 Discussion

In this experiment we observed that even though it was relatively easy to detect basic movements, combining these movements to form specific gestures was more difficult due to certain aspects we did not originally foresee.

4.3.7.1 *Experiment and system improvements*

One way we have tried to improve our results is to eliminate the use of target sticks and *cookie stretch*-luring. One solution is to design clear markers of the locations to be targeted by the dog when performing a desired action. These markers would allow both the dog and

Table 16. False positives experiment. The up and down gestures triggered much more so than the other, while still being difficult to perform. For this reason they were not included in the repetition experiments.

	Session 1	Session 2	Session 3	Session 4
Dog	S1	S1	S2	S3
Duration	15 mins	60 mins	15 mins	15 mins
Scenario	Stairs, walking crossing street	Stairs, car travel, play	Open, play	Open, play
False Positive	Spin(1) Left Reach(1)	Left sequence(2)	Down (4)	Left Reach(1)
Cause	Collar moved vigorous shake	Repetitive left movements while going down-stairs	Looking up while chasing object	Running and turning

the human to train the desired movements with more precision.

Another area of improvement is the sensitivity to orientation. This detection procedure assumes the sensor remains below the neck when the candidate gestures are being performed. Positioning was particularly difficult in dogs with a smooth coat, (e.g., S3), in which the collar tended to rotate and slide freely. Even though the effect of gravity favors the center position, the sensor still shifted for a non-trivial amount of time as a result of vigorous activity.

4.4 Conclusion

In this study we prototyped an on-board communication system for working dogs based on head gestures detected by an inertial sensor placed on the collar.

In this effort we have taken our second step in tackling gesture recognition separately from everyday activity recognition. In this study, we encountered and recognized some practical difficulties that we had previously not considered. For example, we showed the importance of considering the devices the dog might already be wearing such as a leash, harness or existing collar when selecting the gestures. Because these devices can affect a dog’s behavior, they can indirectly affect the performance of the desired movements and

the gesture selection process should account for this effect.

We have also realized some difficulties in training the dogs to perform the movements for the first time, and how these difficulties can negatively impact the recognition results.

During this experiment we also eliminated consideration for one type of movements (those requiring moving the head up or down) because of difficulties in training and a propensity of dogs to move their head vertically while walking.

In total we found there were substantially more requirements on movements that would make good gestures than we originally anticipated.

Before moving on to consider a new system that can address these challenges, we would like to first summarize the requirements for ideal gestures that we have learned so far. This summary is presented in the first part of Chapter 5.

CHAPTER 5

GESTURE CONCEPTION, SELECTION AND DEFINITION

This chapter summarizes the considerations involved in conceiving, selecting and formally defining new gesture movements for communication.

5.1 Introduction

As we prototyped early gesture recognition systems, we noticed some of the qualities that made certain movements good candidates for gestures. These qualities often involved significant trade-offs with each other. We consider these qualities as bounded by constraints, and so, each quality could not be optimized individually. In this way, the gesture selection problem could be conceived as a constraint optimization problem.

5.2 Gesture selection constraints

When we began studying gesture interfaces, it was our impression that there were three principal requirements, which tend to be present in all ideal recognition systems. These are maximizing true positives (recall), minimizing false positives (precision) and in some domains, maximizing generalizability.

As we attempted to train our first set of gestures in the second study (Chapter 4), it became clear we had more requirements than traditional recognition systems. There were also significant requirements of the gesture set as a whole. We noted that, with the exception of Ashbrook et al., even when considering human gesture recognition studies, these constraints are under-acknowledged or not explicitly stated [39].

In this chapter we study these constrained requirements, particularly physical ease, conceptual ease and present results on the propensity for false positives along four candidate gestures. For the purpose of false positive analysis, we reverted to offline analysis of recorded data to be compared against a stream of everyday activity.

5.2.1 Seven requirements

Our seven requirements stem from considering differences between dogs and human users. For example, unlike humans carrying their phones inside their front pocket to avoid accidental calls when sitting (false positives), dogs are not expected to modify their behavior to avoid triggering a certain action. Similarly, unlike humans attempting to speak clearly and slowly to increase the accuracy of a speech recognition system, dogs are not expected to modify their behavior to increase recognition (true positives). This observations was true even if we considered that our participants would repeat their behavior when interacting with tangible interfaces (Chapter 3) depending on the feedback received (or lack of it). Nonetheless, our system could not expect all working dogs to behave in this way.

Similarly, humans can benefit from receiving continuous feedback while performing a given action. For example, through verbal instructions or demonstration. In contrast, feedback for dogs was essentially limited to a binary yes or no response (e.g., reward or no reward).

For the benefit of our experiment participants, we also had to ensure that new candidate gestures did not affect a previously learned behavior. Overall, we relied on movements the dog could perform in-situ as opposed to gestures that required displacement of the dog. That is, we avoided gestures that required the dog to walk or run forwards or backwards. These requirement constraints are graphically summarized in Figure 22.

The requirements are the following:

1. Generalizability across subjects: We should not rely on gestures that can only be performed by a single participant and considered exceptional even among dogs of a given occupation.
2. High true positives (System sensitivity): The system must detect the gesture correctly each time if it is performed correctly.
3. Low false positives (System specificity): The system should minimize alerts when no gesture has occurred.

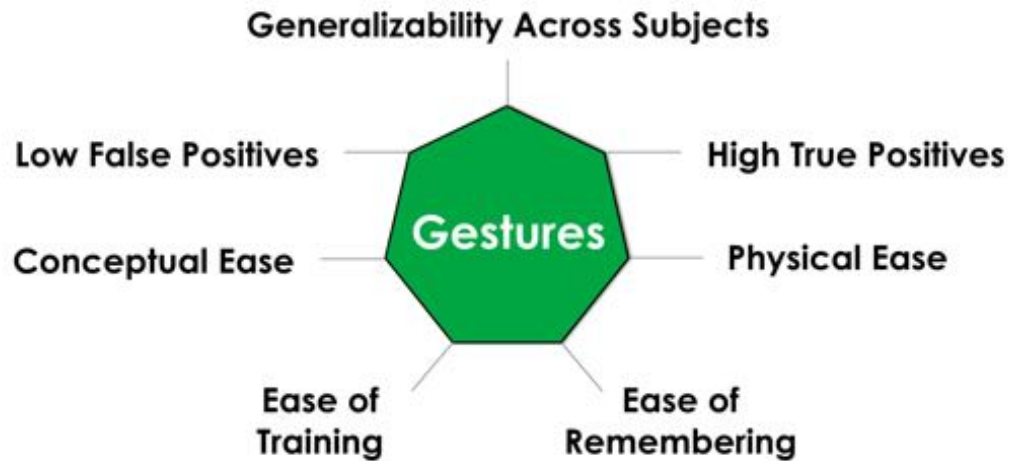


Figure 22. Ideal characteristics of gestures for detection. For canines, the ease of training aspects is of utmost importance.

4. Conceptual ease: Dogs must understand the gesture. For example, repeating a gesture an arbitrary number of times would likely be conceptually difficult and not meet this criterion.
5. Physical ease: The dog must be able to perform the gesture. For example, a back flip would not meet this criterion.
6. Ease of training: A human being must be able to train the gesture.
7. Ease of remembering (Memorability): The dog must be able to remember the gesture after the training phase.

5.2.2 Plotting the requirements

Having identified the set of seven requirements, we could now compare the benefits and drawbacks of each type of movement using data from our previous study. Although not all the axes are quantifiable yet, our experience has allowed us illustrate them in provisional form (Figure 23). Even if the given scores can change when testing a greater number of dogs of different backgrounds, they represent our understanding at the end of the second study and the beginning of the third.

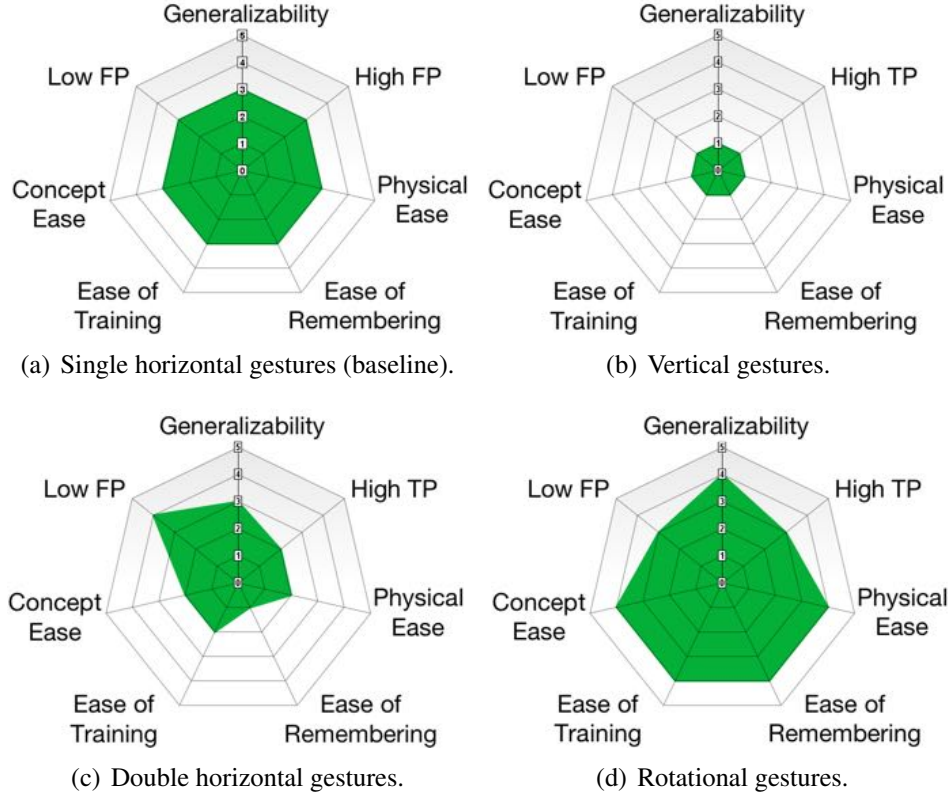


Figure 23. Gesture constraints relative to single horizontal gestures, which serve as a baseline.

5.2.3 Separating the constraints

We finally realized that out of the seven requirements detailed earlier in this chapter, the first three that are typically associated with recognition systems are strictly related to the system not the gesture movements.

The other four requirements we discovered were a property of the gesture movements chosen, rather than the system. For this reason, we decided to illustrate them as two separate sets of requirements that nevertheless influence each other (Figure 24).

5.2.3.1 Gesture selection

The gestures that remained after considering the criteria above were *right reach*, *left reach*, *spin* and *twirl* (Figure 25), while gestures requiring movement up or down were discarded, along with all *double reach* sequences (Table 5.2.3.1).

As explained earlier, reaching left and right were inspired by the movements used to

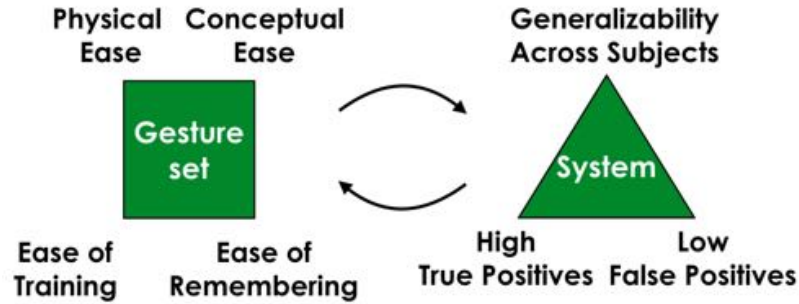


Figure 24. Requirements for ideal gestures separated between gesture requirements and system requirements.

trigger wearable tangible interfaces, but we removed the repetition aspect from previous experiments. *Spin* and *twirl* are 360° clockwise and counterclockwise rotations. These movements are already taught to some dogs, but not performed regularly enough to be discarded at this stage.

Table 17. Definition of each gesture movement under consideration.

Candidate Gestures	Description of gestures
Spin	Clockwise rotation of 360°
Twirl	Counterclockwise rotation of 360°
Right reach	Reaches to right ribcage and return.
Left reach	Reaches to left ribcage and return.



Figure 25. One participant performs the twirl (top) and left reach (bottom) gesture.

5.3 Protocol and participants for false positive study

Up to this point, we have analyzed recognition of certain movements and analyzed their suitability as gestures for communication along a set of seven requirements. Nonetheless, it was now important to conduct an experiment focused on estimating the propensity for spontaneous false positives in each movement. To do this evaluation, it was first necessary to record accurate examples data from each dog performing the candidate gestures.

We now describe how we trained, prompted and recorded the gesture movements we examined in this study with a dog having no previous experience performing gestures on cue.

To avoid training a dog to perform a movement that would ultimately be undetectable from an everyday movements, we first tried to “lure” them into performing each candidate gesture. As we described in Chapter 4, luring is a technique by which dogs follow a target object (e.g., treat or toy) to perform an action [63]. Even though we no longer used food as a target in luring, as in our second study, we ultimately realized that readings from lured actions were more representative of the trainer’s arm movement than the dog’s performance, and hence could not be used as a stand-in for the dog performing the movement on his own. Still, luring was valuable for dog training, but we did not record these instances as gesture templates.

Instead, we had to ensure dogs could learn to offer these gestures after being given a visual or verbal cue. Our first participant had limited previous experience with wearable tangible interfaces and would not offer actions like “reach left” or “reach right” spontaneously. Our second participant had experience with the gesture movements, but performed them in broad undirected ways when lacking a precise target. We realized that even though a gestural system no longer required a dedicated interface for each alert, it was still necessary to have a visual or tactile target while the dog was learning the movement and until a recognizer could provide feedback upon successful completion.

Although we experimented with auditory feedback (throughout and upon completing

a gesture), we believed that using a simple harness-based two-target system was enough to obtain a degree of precision comparable to the tangible interfaces. Although the harness provided a substantial improvement for training, we later learned that the precision of the gestures could not approximate the precision of the physical interactions with tangible interfaces without also relying on auditory feedback. We provide further insight into this phenomena in the conclusion section.

This harness consisted of two bright colored targets on each side (Figure 26). We built the targets out of bright yellow 3.81 cm (1.5 in) diameter balls to make them easier for the dogs to see [64]. Originally, we used a dark target against a dark background (the harness) as a marker, but that was harder for the dogs’ to locate as it did not provide enough visual contrast [65].



Figure 26. Participant wearing the instrumented Julius K9 harness and Shimmer 3 before a training session.

5.3.1 Participants

For this study, one human trained two dogs using positive reinforcement. One dog, a two-year-old retriever cross (R1) with assistance training, had no experience with wearable gesture interfaces and was trained to use them exclusively for this experiment. A seven-year old border collie (BC1), had three years worth of experience with wearable interfaces.

5.3.2 Dog training protocol

When dogs attempted to perform one of the requested movements they received a food reward (1 cm sized treat) [66]. When they could perform the movement correctly at least 65% of the times asked, the reinforcement schedule was decreased to one treat per successful completion [67]. Throughout this process the human also provided prompt feedback

with a click sound. For training *spin* and *twirl* we relied on luring at the early training stage before transitioning to a subtle hand signal and verbal cue [68].

5.4 System and equipment

In addition to the two-target dog harness used for training, the main piece of equipment used for this study was a commercially available inertial sensor platform, the Shimmer 3, by Shimmer Sensing Inc. This unit consists of a nine axis sensor, including three axes of accelerometer, gyroscope and magnetometer. The sampling frequency was set to 51.2 Hz.

As with previous sensor platforms, we selected the Shimmer 3 due to its light weight and small size (51 mm x 34 mm x 14 mm) compared to sensors with similar capabilities. Considerations of weight remain extremely important because heavy objects might obstruct the intended movements. The weight of 28.3 grams, is significantly below the maximum weight guideline (4% body weight) for wearables in Animal–Computer Interaction [69]. Finally, we used a two-pocket harness to place a mobile phone for longer-term wireless recording of everyday movements such that the resulting data matched our target scenario, which at the time involved wireless transmission of data.

5.5 Detection and classification considerations

We originally approached our goal as a traditional classification problem. We recorded examples of dogs performing a given set of movements when prompted by their trainers, for use as ground truth in supervised learning.

Because not all dogs could perform the same set of movements with verbal-only cues, there were insufficient examples to properly train a statistical classifier at this time. Even before we could use such a classifier, we would require a way to segment movements of interest from all other possible movements (null class). Similarly, we lacked the knowledge and sufficient data to infer and train the gesture’s state topology in a hidden Markov model (HMM) capable of performing continuous recognition.

Training either type of model also required human labeling of each example, which was particularly difficult because there were no universally agreed upon definitions of where the desired gestures started and ended. In humans, this requirement is met by enforcing a specific definition on the subject performing the gesture (“you must do it like *this* [while executing a movement for demonstration], otherwise it will not count”). With dogs, only discrete feedback (e.g., reward or no reward) could be provided.

Another issue that remained from the second study was that dogs in training tended to maintain eye contact with their human, which created a different head orientation than a dog performing a gesture movement autonomously. The impact of these issues could be minimized with additional training but, as we described earlier, the availability and energy required to train our participants made it undesirable to train candidate gesture movements for months only to realize later they would be undetectable from everyday movements. To avoid this vicious cycle we decided to only train candidate gestures for which we could do the following three items as needed:

- train all of our previous participants (e.g., *roll* would not meet this criteria);
- visually verify gesture completion;
- and/ or arbitrarily increase the precision and consistency

5.6 Inertial data annotation

We did no explicit labeling in this study other than storing each example of a movement in an individual file, but left the definition of boundaries undetermined. The only definition we attempted to enforce on the dogs when performing the movements was that each candidate gesture movement would be preceded and followed by inertial silence. Ultimately, we abandoned this definition too when, despite our best efforts, none of the recorded examples satisfied this condition. In addition, we feared that requiring inertial silence before and after a gesture might limit the potential for multi-gesture sequences to be re-considered in the future.

We collected everyday inertial data over five-hour periods (including walking, playing, or following a human) and compared recorded examples (templates) of these candidate gesture movements against these everyday streams (so called everyday movement libraries) to gauge their viability as in Ashbrook et al. [39]. We also compared these example templates against data sets containing other examples of the candidate gestures. In this way a false positive could be defined relative to the threshold of true positives. Otherwise, an arbitrary unachievable distance threshold could be set to bring false positives down to zero.

Our hope was that, even if these single gestures proved unsuitable relative to the seven requirements, they could still be used as building blocks of successful ones. Due to this need, we decided to remove the requirement of having the dog be still before and after performing a gesture. In this way, any efforts training the dogs would not be in vain.

5.6.1 Orientation correction

Unlike a human wrist-watch used for gesture detection our inertial sensor was attached to the collar and its position at any given time was not constant [39]. As a result, the readings, which are based on an internal reference frame, might not be the same for similar actions.

To decrease collar movement, we tried more than five collars (some commercial and some custom-made), tried superimposing two on top of each other, and even resorted to clipping the collar with hair clips, all to no avail. Because canine skin around the neck is inherently loose, and canine bodies are covered in thick hair, it became close to impossible to prevent the sensor from moving independent of dog movements.

More importantly, even in cases where the sensor had not moved, the angle of the dog's head might vary for the same movement, for example, when maintaining eye contact with the human. The most effective way to address these issues at that time was to transform the coordinate system of gyroscope measurements into an external reference frame relative to earth.

This new reference frame was thus based on the direction of gravity sensed at any given time. We denote the output of a three-dimensional accelerometer as $\vec{d}_{total} = [a_x, a_y, a_z]^T$.

This output contains two components, namely: $\vec{a}_{total} = \vec{a}_{linear} + \vec{a}_{gravity}$. Next, we represented magnetometer readings as $\vec{m} = [m_x, m_y, m_z]$. The first dimension in our new reference frame was given by $\vec{h}_1 = \vec{a}_{gravity} \times \vec{m}$. Our second and third dimensions were given by $\vec{h}_2 = \vec{m}$ and $a_{gravity}$, respectively. We then normalized each vector to obtain unit vectors of direction $[\hat{h}_1, \hat{h}_2, \hat{a}_{gravity}]$. With these unit vectors, we finally had all the necessary components to create a rotation matrix that was applied to each incoming gyroscope reading.

$$\begin{bmatrix} \vec{h}_1^T \\ \vec{h}_2^T \\ \vec{a}_{gravity}^T \end{bmatrix} = \begin{bmatrix} \hat{h}_x & \hat{h}_y & \hat{h}_z \\ \hat{m}_x & \hat{m}_y & \hat{m}_z \\ \hat{a}_{gx} & \hat{a}_{gy} & \hat{a}_{gz} \end{bmatrix}$$

5.6.1.1 Obtaining gravity from acceleration

It was crucial to prevent the dog's movement at any given time from influencing the calculation of the gravity vector for the correction described above. The most common method to separate linear acceleration from gravity is to use a low-pass filter. In our case this method is also suitable, although some modifications were necessary. For example, when comparing a single gesture template (query) against a larger data set (database), each must be corrected for orientation before they are compared. Unfortunately, the acceleration readings in the query segment include more of the dog's movement than the static gravity readings needed to correct for orientation. For this reason, if the acceleration value at a given point surpassed 1G (9.8 m/s^2), it was not used in the calculation of gravity (Figure 27).

$$gravity_{est}[i] = \begin{cases} \alpha * gravity_{est}[i-1] + (1 - \alpha) * accel_{raw}[i], & \|accel_{raw}[i]\| \leq 9.8 \text{ m/s}^2 \\ gravity_{est}[i-1], & \|accel_{raw}[i]\| > 9.8 \text{ m/s}^2 \end{cases} \quad (5)$$

The smoothing constant was determined to be $\alpha = 0.99$, while $accel_{raw}$, denotes the raw input and $gravity_{est}$ denotes the filtered output.

To verify its effectiveness, we implemented this correction in an online system and

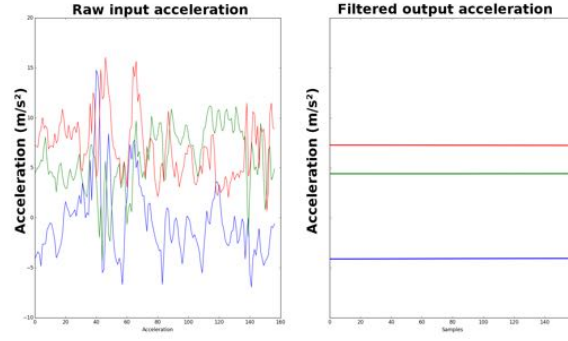


Figure 27. Comparison of a raw acceleration signal of a single gesture and the resulting gravity vector over time. We used these values to correct the orientation on gyroscope readings.

observed that yaw movements relative to earth showed up on the same axes regardless of the orientation of the head or sensor. The benefit of this correction on recognition depends on how much the collar is sliding without the dog moving and even with minimal sliding, the correction allows us to observe a similar signal regardless of whether the head is raised or parallel to the torso.

5.6.1.2 Variance of three-axis norm

We relied on an energy-based approach to segment events of interest from continuous inertial streams. For this type of event-driven approach, the segmentation criterion is perhaps the most important aspect.

The simplest segmentation approach we used was to detect an *event start* when a given intensity threshold was met and an *event stop* when the current reading fell below that same predetermined value. The main drawback with this approach was that zero crossings fell below every threshold. As a result, movements with zero-crossings (e.g., left reach, right reach) were interpreted as multiple segments rather than one. Although this approach can be beneficial in some contexts, it was not suitable for our purposes. We ultimately obtained best results observing the variance of the L_2 norm, $Var(\|gyro_{i:i+n}\|)$ in n samples ($n=40$) when sampling at 51.2 Hz (Figure 28).

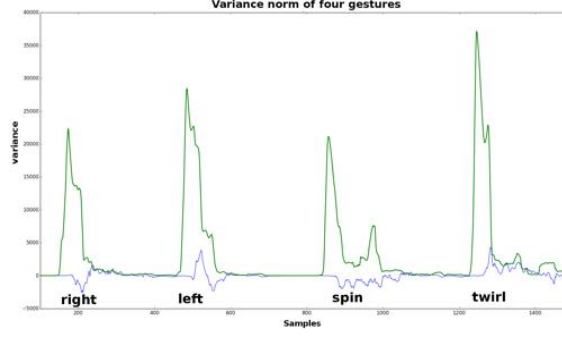


Figure 28. Variance norm in green overlaid over one amplified dimension of the raw signal for comparison.

5.6.2 Event segmentation

From an output like the variance-norm vector a segmentation scheme determined the start and end boundaries of an event. At first, we used a strong threshold that only detected high-energy movements and lost the initial portion of the movements (including potential gestures) whose intensity was below the threshold. When this proved insufficient, we used two sets of thresholds on $var(\|gyro[i]\|)$, one that detected the presence of a movement of interest ($T_{detect} > 4,000$) and two weaker thresholds (T_{start}, T_{end}) to determine the gesture boundaries. This step was crucial, because even a small error in boundary detection dramatically affected the recognition results.

We also placed constraints on the length of the movements of interest (number of samples), intensity of the variance norm and, for *spin* or *twirl* movements, we also placed constraints on the angle traveled. The angle was computed by $\theta = \int |\omega| dt$, where ω is angular velocity at a given time. Because our start and stop criteria allowed for some noise at the beginning and end of a movement, the angles rarely summed up to the expected distances of 180° and 360° for reach and rotation movements. As a result, the only way to distinguish some false positives from *left reach* and *right reach* was through a threshold level that excluded *spin* and *twirl*. For this reason, we had to resort to two thresholds $T_{detect-rot}$ and $T_{detect-reach}$.

Based on the types of gesture movements we were studying, we only used the y (roll)

Table 18. Segmentation criteria determined empirically for movements sampled at 51.2 Hz where θ denotes the angle of movement (e.g., an ideal Spin is 360°).

	Start	Reach	Rotation	End
Condition	$T_{start} > 1,600$	$T_{detect-reach} > 11,000$ samples > 60 samples < 140	$T_{detect-rot} > 4,000$ $\theta > 140^\circ$ $\theta < 390^\circ$ samples > 45 samples < 200	$T_{end} < 1,500$

and z (yaw) readings of the gyroscope such that the head orientation moved upwards or downwards while doing a gesture were not a consideration.

5.6.2.1 Distance metric

We employed dynamic time warping (DTW) to compare signals against segments of a stream. The resulting distance was divided by the sum of the length of each signal to account for the fact that longer sequences have more chances to diverge.

We tried to impose a locality constraint of $w=10$ to avoid pathological warpings. It had no net effect because comparing signals of different lengths imposed a constraint ($warping = \max(length(a - b), w)$) which always yielded $length(a - b)$. We empirically determined $dtw.dist = 50$ from observing the recordings to be the threshold for identifying a movement as an example of a given class. We additionally tried to add several features over time, such as differences between each pair of samples, displacement up to a certain point, but these did not provide any benefits in minimizing distances at this stage.

5.6.2.2 Parameter tuning

To verify the event segmentation and distance metrics, and to obtain the parameters listed above, the first step was to collect small data sets of all gesture movements being performed consecutively. These typically had the form of *right reach*, *left reach*, *spin*, *twirl* at fixed intervals. These data sets were compared segment by segment against two templates of each individual gesture (Figure 29).

The expected result was that during the occurrence of a gesture movement (e.g., *right reach*), its stored template(s) resulted in the smallest distance while other candidate gestures

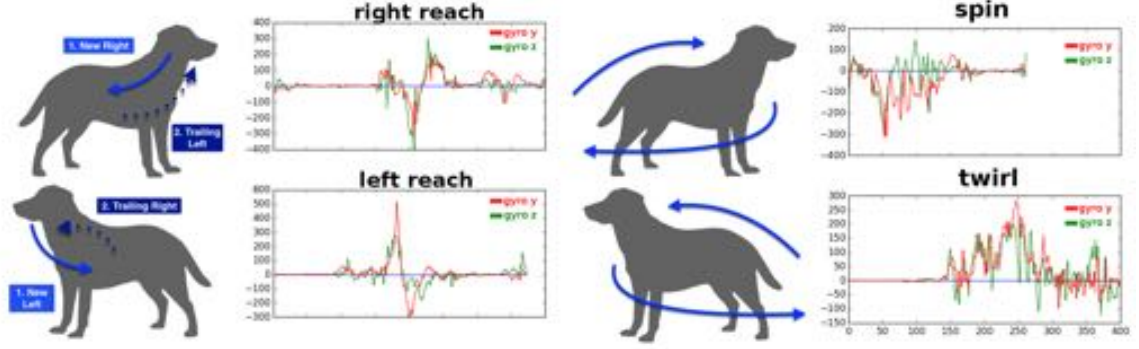


Figure 29. Example of gesture templates used for comparison against streams of data.

(*left reach*, *spin*, *twirl*) resulted in larger distances (less similar).

Figure 30a shows the result of evaluating each training data stream against two templates of the same dog performing each of four candidate gesture movements. Our event segmentation criteria then segmented out the area of interest (Figure 30b). Finally, the dynamic time warping distance was calculated for each candidate gesture movement (Figure 30c).

In this way we arrived at the true positive distance threshold. A segment below $dtw.dist < 50$ was classified according to the movement gesture having the smallest distance. In Figure 30, the first motion was classified as right (red) while the second was classified as left (green).

To evaluate both true positives and false positives at once, we collected data over longer periods of time (25 to 50 minutes) where the dog would perform everyday movements such as those in walking, running, playing, lying down, drinking water and perform candidate gestures at fixed intervals. We refer to these sets as *interval everyday movement libraries* (iEML) in the results (Table 19).

5.7 Discussion

The results of this third gesture interface study were very encouraging. There were no substitutions between any of the four gesture movements. Similarly, for all the gestures performed as part of the iEML there were no deletions (false negatives). Part of the reason

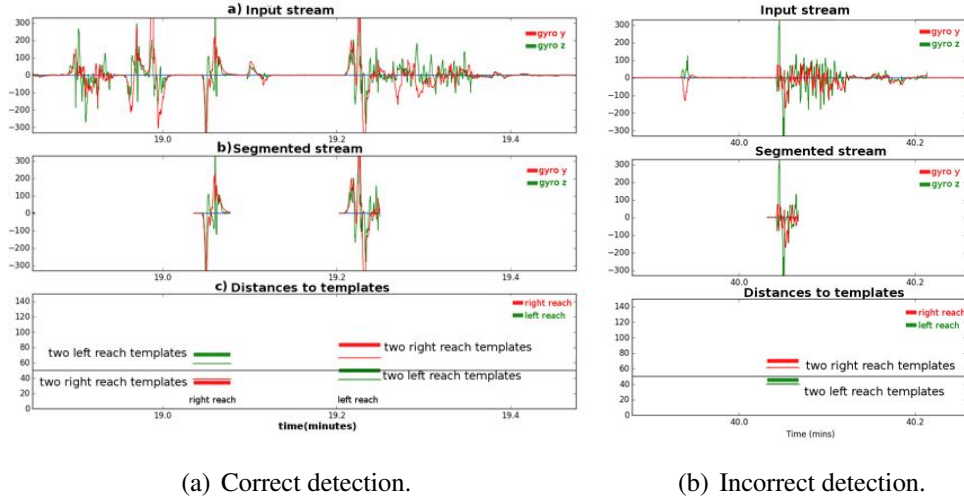


Figure 30. Example result of a right and left reach detected or misclassified in data sets containing other movements.

Table 19. Summarized results for each data set. Some dogs offered gesture movements more than four times.

Dataset	Minutes	Dog	Use	Events	False Pos	FP/hr	Precision	True Pos	Recall	Word Accuracy
iEML1	50	S1	Training	48	1 (left reach)	1.2	80%	4/4	100%	75%
iEML2	25	S1	Training	47	0	0	100%	4/4	100%	100%
iEML3	25	S2	Training	32	0	0	100%	5/5	100%	100%
iEML4	25	S2	Testing	37	0	0	100%	6/6	100%	100%
iEML5	25	S2	Testing	6	0	0	100%	4/4	100%	100%
EML1	305	S1	Testing	50	2 (spin, right reach)	0.4				
EML2	305	S2	Testing	18	0	0				

is that when tuning our parameters, we placed extra emphasis on correct identification because without it, no comparisons to false positives would be possible. In other words, even though our sample size of gesture movements performed (23 examples) is not sufficient to justify broader conclusions, it is a bare minimum to provide a reference for comparing gestures against each other.

There was one case of a false positive motion (rightmost image in Figure 30) that we could not eliminate with the criteria described above. The dog turned his head while looking upwards (most likely at the trainer) and our system detected it as a left reach. The difference from a real left reach was that the z axis had significantly less movement, but we could not codify this requisite in a way that achieved rejection. We surmise a statistical

classifier would be able to encode this separation with enough training examples.

Most of the movements scored as false positives by the metrics occurred due to the spontaneous repetition of the gesture movement requested (for *left reach* and *right reach*). The dogs most likely repeated the gestures because of the lack immediate of feedback. We must note that even though the metric penalizes repetition, for practical purposes, repetition of gesture movements is actually beneficial because it suggests the dog has an understanding of the task and is trying to ensure it is completed correctly.

For *spin* and *twirl*, we expected some false positives to occur because the dogs did perform an equivalent motion while playing. From this experience we found it useful to make the following distinction.

5.7.1 Types of false positives

We have found it useful to make a distinction between two types of false positives, *system false positive* and *behavioral false positive*. The first type are cases where *movement_i* looks like *movement_j* to the recognition algorithm. The second type refers to cases where one candidate gesture movement turns out to be a behavior present during daily living. For example, it might be that certain subjects perform a movement spontaneously (e.g., rotation) before lying down.

Behavioral false positives cannot be eliminated except by redefining the gesture movement in a more specific way. That is, ideally movements that are behavioral false positives can be redefined so that they can be distinguished from their gestural counterparts. The behavioral false positives can be estimated by the human eye while system false positives depend on the classifier.

5.8 Challenges of the gesture set

The current set of candidate gestures required a recognizer to discriminate between mirror images of symmetric movements. This fact led to the poor performance of recognizers that relied on features that described the gesture as a whole but not the order of the movements.

For example, suppose we have the mean of the signal as a feature. This feature would be a poor one for our gesture set because the mean for *left reach* and *right reach* might be the same, but the waveform represents different movements.

The current candidate gestures overlap in definition such that analyzing a small part of a movement is not sufficient for recognition. For example, a rightward peak could be the start of a *right reach*, the start of a clockwise *spin* or the end of a *left reach*. This issue was a problem which we observed earlier in Chapter 4.

The issue of overlap in candidate gestures also affects how the dogs perform the movements. For example, we noticed some dogs start a new repetition before the previous movement is complete. In a way this is an example of co-articulation which we described earlier in Chapter 2.

Finally, our problem area, gestures for communication, required concrete predictions to be provided to the human user. It is not enough to assign a probability for each state like other algorithms. For example, we cannot ultimately say there is 70% chance the dog performed one movement and 30% he performed another. The system must emit one prediction as its final output.

5.9 Conclusion

The methodology we have presented was suitable for analyzing and comparing gestures against every day movements. From our results, we have been able to understand the requirements and constraints of minimizing false positives. We also addressed some practical problems in this area. For example, we illustrated a method for correcting orientation on a dog collar, even for very short segments. We also observed that it is possible for dogs to perform movements on leash without significantly affecting recognition.

Finally, we found four candidate gestures that could be trained, and recognized in addition to discovering a novel way to train them. These gestures were recognized with 75-100% word accuracy and their false positive rate averaged to less than one per hour.

Despite the feasibility testing provided by this study, some challenges remain. The system described in this study required an inordinate amount of ‘hand tuning’ which was acceptable for evaluating gesture feasibility but not for recognition in a deployed system. This system also relied on subject dependent data. In the next chapter, we will describe how we began an to develop a more general recognition pipeline for learning everyday movements.

CHAPTER 6

RECOGNITION OF MOVEMENTS IN CONSTRAINED ENVIRONMENTS

In our previous study we concluded with an analysis of candidate gesture movements and how often they elicited false positives during everyday canine activities in a outdoor environment. To complement this study, we thought it pertinent to perform a similar study of everyday canine movements, but relying on a statistical learning classifier rather a template-matching approach. Nonetheless, recording everyday dog activity in an outdoor environment presented many logistical challenges and we were ultimately unable to perform this type of study. Still, we were able to perform a similar analysis on data from a separate study intended to monitor untrained canine behaviors associated with anxiety, and gained useful understanding for our purposes.

In this way, we could better understand how prevalent some potential false positive movements were in everyday canine behavior more than in our previous study.

6.1 Introduction

Our next goal was to develop a robust movement recognition system. This system should be robust by allowing the user to perform each step individually, with minimal hand-tuning and with subject independence. We call the resulting system a ‘gesture recognition pipeline’.

This pipeline was developed during a study to understand canine kinematics in a constrained environment. These movements were outside the candidate gestures and informed our knowledge of potential false positives.

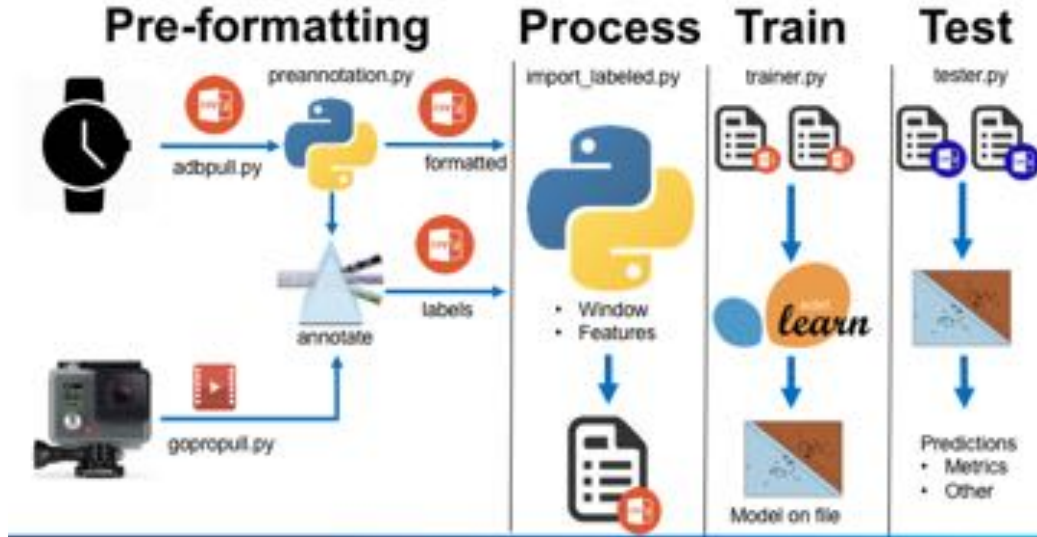


Figure 31. Visual description of the movement recognition pipeline.

6.2 Participants

For the goals of this study, we required participants that would remain active, and perform movements in an enclosed environment, rather than completely outdoors. This indoor movement would allow us to record everyday activity with both video and inertial sensing, something we could not do in the false positive study.

The participants in this study consisted of five police-trained German Shepherd Dogs (GSD) and Belgian Malinois (BM). They ranged between two (2) and five (5) years of age at the time of the study.

6.3 Equipment

For this study, the equipment was different than in previous studies, in that data was stored on the device rather than transmitted wirelessly. This study required devices that could store large amounts of inertial data locally. For this purpose a set of smart watches that could fit to a custom collar was necessary and the *Moto 360* was selected. The *ASUS Zen* and *Samsung Gear* were also considered but ultimately not used. This device, and others considered used AndroidWear OS as its primary operating system.

The hand straps of each watch were removed and replaced by custom straps in order to

form a dog collar (Figure 32). All dogs wore the same watch and collar combination. For this experiment, sampling rate was set to 100 Hz.



Figure 32. Subject illustrating how the Moto 360 was worn on the collar. This dog was not part of this study.

The ASUS Zen was found unreliable in its time stamps, because there were large periods of no data (gaps) in some occasions. The Samsung Gear, on the other hand, had a form factor that a subject expert deemed inappropriate for the participants due to its shape causing signs of discomfort in dogs. As a result, the data collection pertained only to the *Moto 360* (2nd generation) as described above.

The dimensions of the *Moto 360* were 1.6 in x 0.5 in x 0.8 in (4 cm x 1.27 cm x 2 cm) and it weighs 11.2 ounces (317.5 grams).

One *GoPro* camera was attached from the top of the enclosed rooms to achieve a wide view of the dogs horizontal movements during recording sessions.

6.4 Collecting training data

For the purposes of the original study, each dog was placed in a enclosed kennel for a span of 10 minutes. Usually these kennels consisted of fences, but in other occasions the walls did not allow the dogs to see outside the kennel on all directions. During this time dogs were not instructed to perform any specific movement and were free to move or rest.

Each session was annotated by a single human annotator using the *ChronoViz* annotation toolkit. *ChronoViz* is described as a tool to aid visualization and analysis of multimodal sets of time-coded information, with a focus on the analysis of video in combination with other data sources.

Each session was video-recorded and imported into a ChronoViz project along with the corresponding inertial data. For accurate annotations, the video had to be synchronized to the inertial data using common reference points. In the present case, these points corresponded to the human clapping five times with the device in hand. The times where the human was holding the device for synchronization purposes had to be labeled and removed from the data set accordingly. Our experience has shown that despite the small amount of this type of data it is enough to negatively impact our classification.

6.5 Detection and classification

The original movements of interest in this study were behaviors associated with repetitive behaviors, rather than candidate gestures. These movements were clockwise rotation, counterclockwise rotation, flank sucking and a variety of other activities that could cause a gesture classifier like the ones described in earlier chapters to trigger accidentally. Because of the low prevalence of these movements in the data collected we focused on just four basic movements. These movements were *clockwise rotation* labeled as *disp+* (similar but not equivalent to *spin* gesture), *counterclockwise rotation* (similar but not equivalent to *twirl* gesture) labeled as *disp-*, *no displacement* labeled as *nodisp* and a class for other movements labeled as *dispothor*.

6.6 Classifier and feature selection

For this study, we assembled a recognition pipeline that allowed evaluating different classifiers in the final stage. In total, we evaluated the performance of four classifiers. They were support vector machines (SVM) with radial basis function as kernels, stochastic gradient

descent (SGD), standard decision-trees and random forests.

Our feature set consisted of the mean and standard deviation of a windowed signal along each axis after the orientation correction was applied. This set of six features was used to characterize each window. For this study focused on repetitive everyday behaviors the window size was varied between one (1) second and four (4) seconds. All classifiers performed better with the four second window, and this is the evaluation we present below.

6.7 Evaluation

For the evaluation procedure we first performed k-fold cross validation on the data pertaining to all subjects. To avoid known issues pertaining to the same portion of a window being assigned to both the training and testing folds in cross-validation, we did not use overlapping windows during this study [61].

Because each session consisted of 15 minute intervals, they amounted to roughly 60 minutes worth of data, although, as is typically the case, most of this data pertained to the dog performing other activities than those of interest.

For the final evaluation we performed leave-one-subject-out (LOSO) validation.

6.8 Results

We begin by summarizing the class weighted F1 score for each of the four classifiers we examined using five fold cross validation were the following (Table 20).

Table 20. Preliminary analysis of combined data-sets through five-fold cross-validation.

Classifier Type	Five-fold cross-validation score
Support vector machine	72%
Stochastic gradient descent	65%
Two nearest neighbor	82%
Decision tree	75%

We also compared recognition scores on hidden Markov models given the raw window data, but the results did not differ from the techniques presented in this chapter.

Nevertheless, a more accurate picture of the performance of these classifiers can be obtained from performing leave-one-subject-out cross validation (Table 21).

Table 21. Leave-one-subject out cross validation F1-performance metric.

Session	SVM	SGD	kNN	Decision Tree	Random Forest
Dog1	0.5	0.47	0.56	5.0	0.7
Dog2	0.52	0.69	0.67	0.61	0.68
Dog3	0.48	0.52	0.38	0.37	0.71
Dog4	0.75	0.84	0.85	0.87	0.88
Dog5	0.38	0.46	0.54	0.51	0.6

For ease of discussion, rather than discussing each session one by one, we will aggregate the results across several dogs to form individual confusion matrices for each classifier.

As we will observe from the confusion matrices, most of mis-classifications occurred between the null-class of other displacements (*dispothor*) and the other three classes.

6.8.1 Support vector machine

We began our exploration with an analysis of the F1-performance metrics results based on a support vector machine (SVM) classifier with a radial basis kernel function (Table 22).

Table 22. Confusion matrix for the SVM classifier.

	disp_other	disp+	disp-	nodisp
disp_other	30	15	2	0
disp+	35	43	4	0
disp-	2	2	4	0
nodisp	24	3	1	0

The majority of the confusions or collisions arose between clockwise displacement (*disp+*) and other displacement (*disp_other*). There was similarly confusion between no displacement (*nodisp*) and other displacements (*disp_other*).

6.8.2 Stochastic gradient descent

We continued our analysis with a classifier based on stochastic gradient descent. For this study the maximum iteration was set to $max=30$.

This stochastic gradient descent classifier performed slightly better than the SVM classifier (65%). In particular, it was not as affected by the class imbalance as the SVM classifier.

Nonetheless, there was still substantial confusion between clockwise rotation (*disprot+*) and all other movements.

Table 23. Confusion matrix for the SGD classifier.

	disp_other	disp+	disp-	nodisp
disp_other	9	39	0	0
disp+	3	78	1	0
disp-	1	7	0	0
nodisp	23	16	0	12

6.8.3 K nearest neighbors

Third, we considered a nearest neighbor classification system with $k=2$ neighbors.

The nearest neighbor classifier had the greatest confusion between *no displacement* and *other displacement*.

Although it performed better than the SVM classifier, it did not improve on the results of the stochastic gradient descent.

Unlike our earlier study with dynamic time warping and kNN (Section 5.2.3.1) this test did not compensate for the time component and we believe that is the reason that performance decayed considerably. The results of this test can be observed (Figure ??)

Table 24. Confusion matrix for the kNN classifier.

	disp_other	disp+	disp-	nodisp
disp_other	15	17	12	4
disp+	22	48	10	2
disp-	1	2	5	0
nodisp	30	2	5	15

6.8.4 Decision trees

We then considered a decision tree classifier because of the ability of trees to overcome differences in the scales of each of the features. For this study, the maximum depth was set empirically to ten ($max=10$) to avoid the possibility of over-fitting.

Overall we can see that the decision tree performed better than the SVM, SGD and kNN based classifiers. For example, we can see that the confusion between clockwise rotation (*disprot+*) and other displacement (*dispothor*) was significantly minimized.

Table 25. Confusion matrix for the decision tree classifier.

	disp_other	disp+	disp-	nodisp
disp_other	21	14	11	1
disp+	7	64	9	0
disp-	1	2	5	0
nodisp	29	7	5	8

6.8.5 Random forests

Finally, we concluded our analysis with a random-forest based classifier. The parameters used for this evaluation were $n=100$ estimators and a maximum depth of twenty ($max=20$). The class weights applied were balanced relative to their prevalence in the data set. The random forest achieved the best performance among the classifiers we tried.

We can see in Table 26 in this case that the confusion between no displacement and other displacement was not as low as SGD, and did not perform as well in overall F1 score.

Table 26. Confusion matrix for the random forest classifier.

	disp_other	disp+	disp-	nodisp
disp_other	36	9	2	1
disp+	20	60	2	0
disp-	3	0	5	0
nodisp	33	4	1	14

6.9 Discussion

Similar to our earlier experiment with gestures, all four classifiers analyzed tended to favor predicting the null class, even when the cost function was adapted to take class weights into account.

Because of the lack of sufficient annotated data, there was a decrease in performance when generalizing the classification to subject-independent data. The lack of annotated data resulted from two factors. First, there were multiple definitions of the types of repetitive behaviors involving rotations, even among domain experts (e.g., veterinarians). Second, the influence of the legs on the head movement, and how these movements would be labeled, had up to this point been ignored.

Even though this study was not aimed at gesture recognition, the importance of leg movements on the head was a crucial insight for the purposes of developing a wearable gesture interface. For example, two of our candidate gestures in Chapter 5 (*spin* and *twirl*) involve significant body movement with the legs, while the others (*right reach* and *left reach*) do not. As a result, of the differences between head and leg movements our data was annotated relative to the movement of the head in some cases and relative to the body in others. This was also the case in our previous gesture studies. This distinction turned out to be a significant problem because dogs heads, even during rotations, tend to move separately from the body. These inconsistent annotations limited our effective training data (data that did not contradict each other) and ultimately hindered the performance of our classifiers.

Finally we note that the main axis of interest (yaw) for these everyday activities was the same as for our gestures. Recall that in an earth-referential frame yaw is defined as a rotation about the axis of gravity vector. Nonetheless, our correction depended on accurate magnetometer readings, which assumed the sensors were not under the influence of a nearby magnetic field. For this reason, we decided to avoid using the orientation correction mechanism in future studies.

6.10 Conclusion

Despite its inherently different goals, in this study we learned several important lessons about recognizing canine movement.

First we learned that inconsistent annotation labels led to sub-optimal results in the evaluation stage. Second, we noticed that sometimes the dog could ‘rotate’ with their body, but not necessarily their head. In other words, their heads remained fixed looking towards a particular location. Interestingly, we would later observe this behavior as well in dogs performing gesture movements, and the knowledge gained in this study was crucial to account for it.

Finally, we noticed that due to the shifting magnetic field, the orientation correction did not produce the same results we achieved in earlier experiments in the laboratory.

To alleviate these issues discovered in the present study, we began our final study described in subsequent chapters, with three key improvements. First, data was to be annotated programatically. Second, annotations would distinguish between head or leg movements. One would correspond to the head (neck) movement while the other corresponded to the body (leg) movement.

Finally, the low-level kinematic movements would also be classified on a sample by sample basis. This scheme would allow for an expert to provide quick input into labeling decisions without having to analyze raw data. Finally, it would also allow for quick debugging of the classifier in case its predictions degraded in a new scenario.

As will become evident, these were the three remaining pieces necessary to achieve our desired system.

CHAPTER 7

WEARABLE GESTURE RECOGNITION SYSTEM

7.1 Introduction

In this chapter we synthesize the knowledge from four previous studies to construct a new wearable gesture recognition system. We describe the implementation of this system from annotation to classification.

We also developed a system to obtain parameters from a human expert and programmatically use these parameters to apply labels to inertial data. The labels corresponded to low-level individual movements and to groups of movements as well. We used a set of seventeen features along with the resulting labeled data to train a set of classifiers that can recognize movements of interest.

7.2 Participants

For this study, we had four dogs (BC1, BC2, BC3, L1) performing several repetitions of each of the four gestures. They were three border collies (BC1, BC2, BC3), and one retriever (L1). L1 had experience with gesture interfaces, while B2, BC3 had experience with both tangible and gesture interfaces.

BC3 had limited previous experience using wearable interfaces (either gesture or tangible) and was also the youngest of the group. Nonetheless, there were no issues in training BC3 to perform the actions to participate in the experiment.

Table 27. Participant information for the wearable gesture study.

Participant	Breed	Age (years)	Training	Wearable Exp
L1	Retriever	3	assistance	no
BC1	Border collie	7	assistance	yes
BC2	Border collie	6	assistance	yes
BC3	Border collie	1	none	no

7.3 Equipment

For the wearable equipment in this study, we returned to the equipment used in Chapter 4, but it was used in a slightly different manner. It consisted of the same Shimmer 3 sensing platform sampling at 8 Hz. In this experiment, the orientation was changed to be parallel to the collar to minimize the amount of force the neck exerted on the device and prevent it from falling off. This change led to *pitch* and *roll* directions corresponding to different axes than in previous studies. We also constructed a new fabric enclosure to hold the sensor in place in its desired orientation (Figure 33).



Figure 33. Final enclosure and orientation to hold the shimmer sensor parallel to the collar.

In addition, we added a 6ft (1.8m) by 8ft (2.44m) track, which we describe later in the protocol section, to simulate a sidewalk. There were four traffic signs, whose purpose will also be described below, each consisting of a different shape and color. They all had a uniform height of 29 in (73.66 cm) and a weights of 6 lbs (2.7 kg). They were traffic signs corresponding to symbols representing *Yield*, *Stop*, *One Way*, and *School Crossing* in the United States.

7.4 Data collection

After our third study collecting individual examples of gestures to perform a similarity search (false positive study in Chapter 5), we decided that to improve data collection by recording it in a more ‘realistic’ scenario.

At first, we envisioned this scenario would be identical to the system evaluation scenario we would employ upon completion. In this case, the dog would walk along with the human trainer, preferably outdoors while detecting predetermined objects used as cues to perform the gesture. Ultimately, it was not possible at this stage to collect this realistic type of data, due to several logistical difficulties we will describe. These difficulties were ultimately left unresolved and addressed later on to the annotation stage. Nevertheless, in the interest of completeness, we describe our decision process below.

7.4.1 Selection of objects to alert on

Our protocol called for a set of objects that the dogs would be required to detect by performing a candidate gesture to alert the human. The first objects considered for this purpose were toy balls of different colors. Nonetheless, alerting on a colored ball might lead dogs to interpret it as an object of play, and interact with it as such. We then considered employing a set of cones of different colors. Unfortunately, we had no assurance that we could find four colors (one for each gesture) that could be distinguished by canine vision [64].

Afterwards, inspired by our assistance dog scenario, we thought of using small-scale traffic signs. We iterated through different materials to construct the signs. At first they were completely plastic, then paper-based, and ultimately we settled on a combination of plastic signs with a metal stand that brought the sign to (roughly) our average dogs’ gaze level (Figure 34).

7.4.2 Training to alert at sign

The next step was to train the dogs to perform the gesture movements when encountering the small-scale traffic sign. Despite our best efforts, the reactions of our first three dogs was



Figure 34. The final objects consisted of small scale traffic signs.

to, understandably, interact with the sign(s) directly, mostly by pushing them with their paw. Two of the dogs with agility competition experience tried to place their front paws on the metal base due to previous experience with a task known as ‘foot targeting’. Once they learned that a gesture movement was expected, rather than a physical interaction with the sign, dogs seemed to increasingly ignore the symbol altogether and focus exclusively on the nearby human. We then thought it necessary to develop their interest in the signs themselves by rewarding a ‘nose touch’ to the center of the symbol and eventually rewarding only the performance of the gesture movement in front of the traffic sign. As might seem evident from the description above, there were some intrinsic difficulties with this training process. As we mentioned, the dogs originally had no interest in the traffic signs themselves compared to the interest they had in the humans. For this reason, the value of the traffic signs had to be trained and conditioned. To make matters worse, once we had succeeded in developing some interest, we asked them to perform gesture movements, without realizing that performing the movements required them to take their gaze away from both the human and the traffic sign symbols. Despite these difficulties, we were able to train our first dog to perform two gesture movements (one for each of two traffic sign symbols) in a laboratory setting. We then moved to indoor training outside the laboratory.

For out-of-lab training we wanted the dogs to perform gesture movements without stopping their walks shortly after encountering the traffic sign. Instead, our training outside the lab suggested that our participant dogs wanted to first alert the human about the presence of the sign by stopping their walk before performing any action. If the human acknowledged their stopping, they might then perform the desired gesture movements, otherwise they would keep walking with their human companion.

With this in mind, we finally began outdoor training. When we trained outdoors, the dogs seemed to get overwhelmed by the sounds, scents and pedestrians they encountered along the way. Outdoor locations also tended to have multiple pedestrians which came in-between the dog and the camera used to record the training sessions for later analysis. Due to these issues we decided to move the ‘out-of-lab’ testing location to an indoor environment (Figure 35).



Figure 35. Onleash training performed indoors.

Our participant dogs were not active guide dogs. As such, if they were on-leash, as we proposed in earlier studies, their behavior would always defer to the human. For example, their walking speed would mimic the humans’ speed and depending on how the human handled the leash or how much clearance they had, they might not perform a given gesture movement. As a result, we decided to allow dogs to perform gestures off-leash in an indoor area to simulate the dogs independent movement.

We made some significant progress by allowing the dogs to first alert the human of the presence of a traffic sign by stopping abruptly in front of it. To do so, they tended to perform a nose touch movement, which had been trained at an earlier stage.

7.4.3 Training to discriminate between sign symbols

We had four signs that the dogs would need to recognize (Figure 34). It was then necessary to train the dogs to discriminate between each of their symbols. Over time the ‘nose touch’ became a required part of the protocol because it guaranteed dogs looked at the traffic sign before performing the gesture. If they instead ignored the traffic sign altogether, it might suggest they did not look at it, and discrimination between each traffic sign would be unlikely. The nose touch also made the start of each gesture movement consistent for each dog. For example, before the nose touch, some dogs would not face the traffic sign, instead they continued to move in arbitrary directions (e.g., sideways).

With the new concessions (indoors, off-leash, nose-touch scenarios) we were able to train one dog who, in the same session, could distinguish between two sign symbols in a lab scenario. Unfortunately, as we mentioned earlier, this performance decreased outside the lab and was more difficult to replicate in subsequent training sessions.

Some of this behavior was retrospectively understandable. Unlike guide dogs who are taught to react to environmental cues, our dog participants were accustomed to strictly work with human cues. The only previous experience any of our participants had with environmental cues was one of the dogs training to search for a scent and alert when it was found.

Instead, contrary to their previous experience, we wanted the dogs to react to an environmental (non-human) cue that was predominantly visual as we expected a guide dog to do.

7.4.4 Scent training

At this point, it became necessary to contact two expert guide dog trainers to learn the fastest way to achieve the necessary behavior (visual discrimination of symbols). Our consult was very insightful and yielded unexpected information. These trainers explained that even guide dogs who alert to apparent visual stimuli rely in fact, at least partly, on the scent of the objects they alert on, and it was important for them to build a scent association between the cues.

We tried to incorporate this idea into our training sessions with traffic signs. At first we made scent containers with essential oils, one for each movement-sign pair and trained one dog to perform them upon smelling the scent in a given container. Despite achieving a near perfect initial performance, the results were hard to replicate when changing testing locations. This might be a result of a dispersion of the scent in between testing sessions. As a last resort, we finally tried using traffic signs that were each constructed from materials that smelled differently (plastic, metal, wood, etc.).

Ultimately, we decided that although it was possible to teach dogs the discrimination task, our limited time with each dog (one day a week), was not sufficient to achieve the desired behavior. Instead, we could maximize our available time by focusing on collecting movement data using whichever method was most effective and postpone the discrimination task to a later time.

7.4.5 Discriminating between gesture movements

After we decided to focus on having the dogs perform the gesture movements strictly for collecting training and testing data, we allowed the human to cue the dog on which movement to perform.

Our first two dogs had previous experience discriminating between pairs of gesture movements. For example, they could discriminate between right rotation (*spin*) and left rotation (*twirl*). They could also discriminate between *right reach* and *left reach*. Nonetheless, they required additional training to tell which gesture movement to perform when

considering the task of discriminating all four gestures at once.

The main reasons for this extra training were two-fold. First, in our experience, tasks requiring distinction between right and left took longer to train than non-directional tasks such as *sit* or *lie down*. We inadvertently compounded this difficulty by having four directional movements to be performed as part of one single discrimination task.

Another layer of complication came from the fact that the movements necessary to achieve the two rotations often contained smaller gesture-like movements. For example, *spin* could be performed by performing multiple *rights* and *twirl* could be performed by performing multiple *lefts*. In other words, half of our gesture set consisted of subsets of the other half.

We noticed dogs would hesitantly begin to perform all movements with a *reach*, then stop to look at the human for validation. Only if the human did not reward them would they proceed to do a full rotation. It seemed they perceived these two types of gesture movements as almost the same and, hence, had trouble deciding which one to perform.

While we had considered the difficulty of each gesture movement in great detail in earlier studies, up to this point we had not considered the difficulty of a *gesture set* as a whole. As it turned out, having gesture movements of both directions in the same gesture set increased the cognitive difficulty beyond what we expected originally.

7.4.6 Data collection resolution

The method we selected to elicit the gesture movements was to perform multiple examples of only one gesture in a given session such that the dog would not need to discriminate between sign symbols or decide between gesture movements.

We no longer had the human walk next to the dog for the following reasons. If the human was behind the dog, the dog would turn sideways to perform the gesture facing the human. If the human stood by the side, the dog would develop a preference for turning to the side in which the human was standing on and not the other. Similarly, directional cues were not as clear when given from one side versus the front. Most of our participant

dogs were previously taught to perform lateral discrimination tasks by facing a human, and consequently their performance was much better when the human stood in front of them. On the other hand, standing side-by-side to the dog caused the humans' opposite side to be completely out of their view.

7.4.7 Protocol

Instead of having the human sideways to the dog, we decided the human should stand behind the traffic sign and have the dog perform the gesture movement in front of them. To facilitate this training we created a track 6 ft by 8 ft (1.83 m by 2.44 m) which would represent a sidewalk and place the traffic sign at one end. At the other end, we would place a mat for the dog to lie on as a starting point. Upon the human's command, the dog would go to the traffic sign, perform the gesture movement, get rewarded and then be led by the human back to the mat (Figure 36). For our participants, the setup can be seen in Figure 37.

Each dog performed at least ten attempts of each one of the four candidate gesture movements during a regular testing session. Two of these type of sessions were recorded per dog.

7.4.8 Remaining difficulties

Despite the careful consideration involved in designing the new protocol, data collection still presented some new challenges.

The main remaining challenges related to the so-called *midas touch* problem, where every movement performed by the dog, even by accident, is potentially part of the gesture movement [70]. For example, most dogs either hesitated while performing the gesture movement or performed the movement multiple times per cue. In addition, for the reach movements, high drive dogs depended on near-immediate feedback once they reached for their side, otherwise they would start to perform the movement again. Unfortunately, based on the position of the human in the new protocol (behind the traffic sign) the head of the dog often obstructed the human's view, making it difficult to judge whether the gesture was

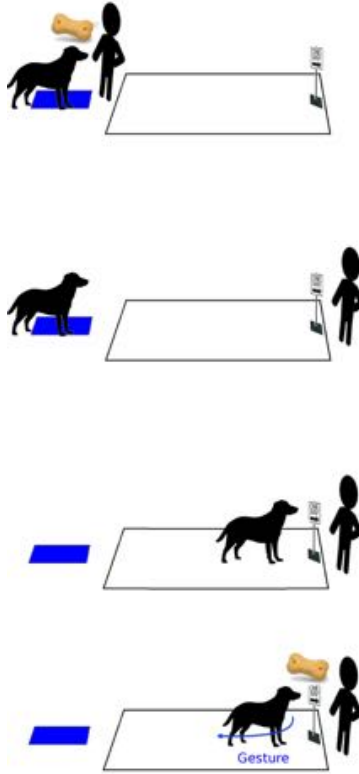


Figure 36. Visual description of the training protocol.

completed or not.

Furthermore, as dogs performed the movement repeatedly they tried to perform the smallest effort movement that would still give them a reward. Despite the nose touch requirement, some dogs still maintained eye contact with a human while performing the gesture movement. Dogs who kept eye contact had a movement path drastically different than that of dogs that did not.

All of these issues come to the forefront, and were ultimately grappled with, during the annotation phase.

7.5 Data annotation

The scheme for data annotation is perhaps the most important aspect of this work.



Figure 37. One participant being rewarded after completing one gesture.

7.5.1 Types of annotations

In constructing our system we developed two types of annotations, the first was focused on the inertial data, while the second was focused on the video analysis.

Originally, we believed video and inertial data annotations were equivalent and could be performed simultaneously, but it turned out that due to often-conflicting requirements, they were best treated separately.

The inertial annotation labels for training and testing required class-consistent labels with precise boundaries to achieve a robust recognition. If a certain movement with a given intensity was labeled one way in one session, it should be labeled that same way throughout that session and all subsequent sessions. Nonetheless, this type of annotation alone was not sufficient, the video annotation labels were also required. These video labels, on the other hand, required human judgment on how many movements in a given session were performed correctly enough to count as a gesture. While both of these annotations (inertial and video) could be used to evaluate detection performance, only the inertial annotation could provide direct input to the classifier.

Unlike our previous system, which was meant to broadly examine the feasibility of the gesture set, this study required us to collect sufficient data to ensure robustness in recognition performance. The logistical difficulties noted in the previous section (Section 7.4.8), prevented us from collecting hundreds of high quality gesture examples. Instead, we focused on collecting a set of examples that were accurately and consistently labeled to serve

as ground truths to train our classifier.

7.5.2 Synchronization of video and inertial data

Early on, we attempted to annotate video and inertial data simultaneously. For this concurrent analysis of video and inertial data we employed the use of the ELAN annotation toolkit (Figure 38).

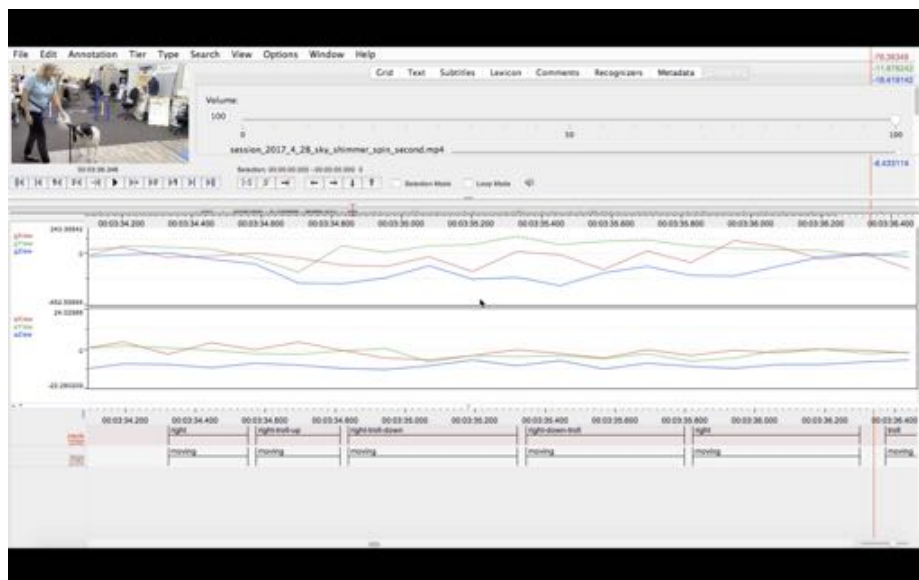


Figure 38. Using the ELAN annotation toolkit to synchronize the streams of inertial and video data.

The video and inertial measurements were synchronized using a common event that could easily be identified both in the video and in the inertial data. The most common way to achieve this, was to record a high intensity event with a distinct pattern. For these experiments our event consisted of six (6) claps while holding the sensor, performed before and after concluding each session. We called each of these events synchronization triggers.

Throughout the early annotation process, we noticed that it was necessary to annotate all movements, not just candidate gestures. As we observed more of the possible movements dogs performed in each session, our annotation labels tended to change over time. We already knew that this lack of consistency could affect the classifier adversely and decided to address this problem before continuing any further.

7.5.3 Inertial data annotation

To achieve consistent labels we realized it was necessary to establish a clearly defined taxonomy of movements. Following the study described in Chapter 6, and the beginning of the present study, we observed the inertial data through the different annotation tool-kits and had gained an understanding of the basic components of each movement. With this understanding, we developed the necessary hierarchical taxonomy of movement.

The first step in the annotation hierarchy was to distinguish between head/neck movements (e.g., nodding), body movements (e.g., ambulating) and the combination of the two. Even though our final classifier relied only on one type of label (neck movements), the conceptual separation of the body label allowed for more consistent annotations. For example, this definition allowed us to distinguish between a dog rotating their head and body simultaneously versus a dog rotating their body while minimizing their head movement because their gaze was fixed on a human. We observed this last behavior in working dogs keeping their gaze set on a fixed target. Whether to annotate this behavior as the same movement as rotating both head and body was a point of debate during the annotation phase. With the new scheme, it was clear that these two movements were to be treated and annotated as distinct from each other. We must also note that making this distinction allowed us to have some of the same benefits as the orientation correction mechanism described earlier (Chapter 5 Section 5.6.1) without relying on an ever-changing magnetic field.

Because we had decided to annotate all movements, the annotation task expanded considerably from only labeling the gesture-like movements. It became clear that dog movements had to be annotated at a very fine grain to capture the amount of detail present in the data, even when sampled at low frequencies (i.e., 8 Hz).

The first step in obtaining consistent annotations was to ensure that each individual sample was labeled correctly. At this stage, it was unfeasible for the human to apply labels to each micro segment, but it was possible to programatically assign them with an annotation heuristic.

7.5.3.1 Multi-axis single-sample labels

The sample-by-sample labels consisted of neck movement annotations. The possible movements of the neck were *roll+* or *roll-* for the *x* axis, up or down for the *y* axis, and right or left for the *z* axis. A final category called *minor* encapsulated subtle movements of the neck detected by the inertial sensor but were too small to be considered a voluntary movement. To ensure consistency we established a threshold of 100 degrees per second (dps) for a sample to be considered as belonging to any one of these classes. The thresholds can be altered slightly (from 50 dps to 100 dps), in our second gesture study we had used 90 dps (Chapter 4), but for our current application it was important to maintain consistency in the gesture definition, so we selected 100 dps for all dogs (Table 28).

Table 28. Definitions for single-axis movements according to gyroscope readings.

Axis above threshold	-100 dps <	None	>100dps
gyro x	roll-	minor	roll+
gyro y	pitch-	minor	pitch+
gyro z	left	minor	right

Each sample along each axis was labeled according to this scheme, such that in one-second samples at 8 Hz we would have 24 sub-labels. These granular labels were not only useful in assigning class labels to larger segments later on, but also assisted the debugging process when the system was not working properly.

7.5.3.2 Single-axis sample labels

These granular axis labels were then assembled into a single label for the three axes of one sample. The precise way to combine them can vary from application to application. In the easiest case, if only one of the tree axis was above the threshold then the single axis label would correspond to that axis. If more than one axis was above threshold, a combined name had to be assigned.

In our case, if the dog's neck was in a flat position, the most common combination in our gesture movements were reflected as *yaw* and *roll* components. The reason can be described as follows. If a dog was moving to either side (from a forward-looking position) a

yaw component. The roll component was also required because, once the neck had moved its maximum amount to either side, the dogs had to either re-position their body by moving their legs or continue a given movement by rotating their neck instead (rolling). For example, if the sample contained both negative *yaw* and negative *roll* movement, and only minor movements of pitch, it would be labeled as *right-roll-*. If it only contained positive *yaw* movement it would be labeled as *right*.

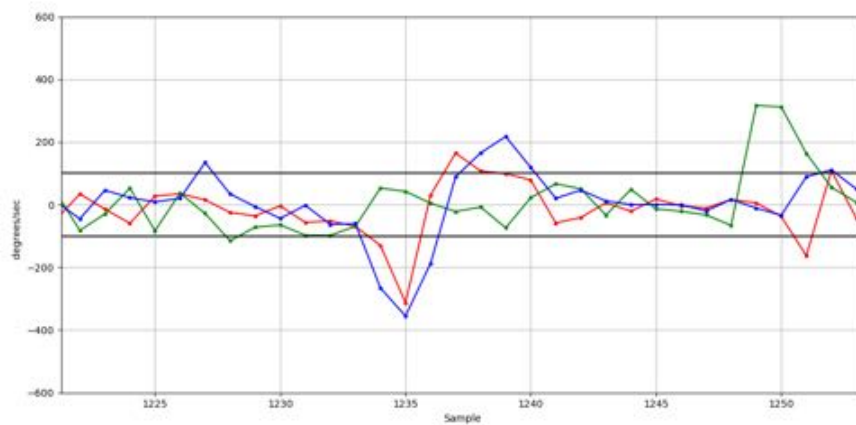


Figure 39. Example of a right reach gesture, the red, blue and green lines represent roll, pitch and yaw respectively.

If, the dog's neck was in a more vertical position, such as when looking up at a human, all gesture movements would be reflected differently. For example, rotations would consist of more *roll* movements than if they were performed from a flat neck position. Reach movements would also be affected because yaw readings would have larger intensities. For example, when the head is already raised the pitch component inevitably increases because the dogs had to reach down before returning to their upward-looking position. Even though we did not intend for dogs to start gesture movements from this position, we decided it was still important to label them (*left-pitch*, and *right-pitch*).

Overall we could describe all movements of interest (gesture and non-gesture) as consisting of one or more of the following single axis sample labels, *right*, *right peak*, *right peak pitch*, *left*, *left peak*, *left peak pitch* or the low-intensity beginning and end of a gesture (*minor*) (Table 29). Ideally, these pitch movements would be further distinguished,

between up and down variants.

Table 29. Summary of multi-axes labels and their names in this study.

Movement	Multi-sample Multi-axes Label
minor on all axes	space
roll+	roll
roll-	t-roll
yaw+	left
yaw-	right
pitch+	up
pitch-	down
yaw+roll+	left peak
yaw-roll-	right peak
yaw+pitch	left peak pitch
yaw-pitch	right pitch pitch

As a historical note, we must remark that this sample-by-sample annotation approach is very similar to the original real-time system we described in Chapter 4.

7.5.3.3 *Single movement annotations*

Because movements consist of samples than one, the next step in the annotation process was to label gross dog movements in a stream of inertial data rather than individual samples. Each possible movement label should describe a class from which the given movement is a member. Since movements consist of more than one sample, the first step in assigning a class label to a multiple sample movement was to determine exactly which group of samples was to be labeled.

One common approach, when appropriate, is to group a fixed amount of samples in a larger sequence. If a group of samples is of a fixed pre-determined length we refer to it as a *window*. Typically, the fixed groupings are applied consecutively until a large sequence is broken down into these smaller groups known as windows. In this approach, the groups of interest are assumed to be homogeneous collections of similar samples, with a similar beginning, middle and end. With this assumption, the window label is assigned by counting

the individual samples present in the window and assigning the group label based on the sample label with highest prevalence (voting). We used this approach in Chapter 6. If, instead, the group of samples is not of a pre-determined length we called it a *segment*.

One common method of grouping and classification is *isolated supervised recognition*. This method relies on the human (not an algorithm) providing the recognition system with the groupings of samples to be recognized as members of a given class [41]. The system should then assign a specific class label to each one of the group of samples provided by the human. Early on, we experimented with a system of this type but its usefulness was very limited [3].

Another possible method, *continuous recognition*, requires the system to perform both steps (grouping and recognition). With this method, the human still annotates parts of interest in a continuous stream of data, but the groupings are not provided immediately to the system. Instead, the human label is broken down into individual sample labels. The system has to then infer how the human grouped the samples in a continuous stream of inertial data through a separate method of grouping. That is, the system has to have its own method of windowing or segmentation. As a result, the parts of interest selected by the human will not necessarily correspond to the same group of samples that the system will window or segment.

Despite this discrepancy, the system must be trained using the class labels from the human annotation, though not the exact sample groupings. Unfortunately, if the class labels do not correspond to the same groupings, their usefulness in learning the correct boundaries will be limited.

In our false positive study, described in Chapter 5, we provided the system with a separate segmentation method based on intensity thresholds [71]. Unfortunately, ensuring consistency between the human label and system labels forced us in the past to hand tune the system segmentation method and limited its generalizability.

Instead, our recognition system in Chapter 6 relied on fixed-length windows, but the

human labels still corresponded to arbitrary length groupings. As a result, the labels with the human groupings had to first be converted to the grouping scheme employed by the system through the windowing and voting procedures explained earlier (Figure 40).

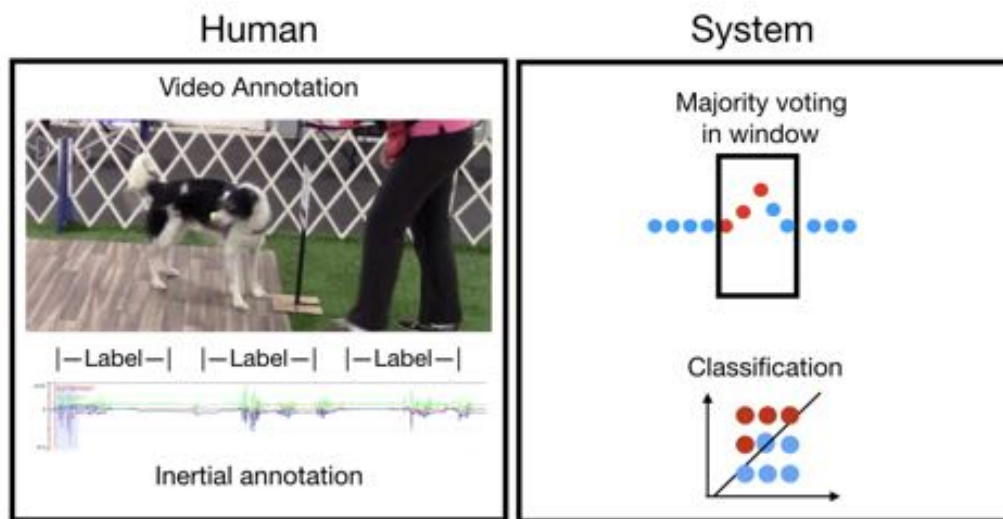


Figure 40. In continuous recognition the human provides annotations but these are re-applied to individual frames or windows.

For example, suppose we have a system that recognizes movements on one-second windows. If the human has labeled a five-second movement of interest as *Label A*, that label would need to be broken down into samples and applied to the equivalent amount samples corresponding to one-second windows (five if there was perfect alignment).

We ultimately defined the following labels for multi-sample single movements: *RightPeak*, *RightPeakPitch*, *LeftPeak*, *LeftPeakPitch*, *Right*, *Left*, *Space*, *Right Spin* and *Left Twirl* as seen in Table 30.

We note that we did not use voting, or required homogeneous groups. If the group was homogenous, the group was labeled with the same labels that composed it. If, instead, the group consisted of different types of single axis samples (non-homogenous groups), we required only one sample of *Right Peak* or *Left Peak* movement for the segment to be labeled as such. We finally modified the definition to use the maximum intensity of each axes on the segment, and if these axes met the yaw and roll criteria for *Right Peak* or *Left*

Peak we assigned the segment label based on that, even if no individual sample was labeled as such.

Table 30. Summary of the set of multi-sample single movements defined for this experiment.

Single Movement Label	Definition
Left	Collection of left movements
Right	Collection of right movements
Left peak	Collection of left movements with roll+
Left peak pitch	Collection of left movements with pitch.
Right peak	Collection of left movements with roll+
Right peak pitch	Collection of right movements with roll-.
Right Spin	Collection of right movements sustained for a set length.
Left Twirl	Collection of left movements, sustained for a set length.
Space	Collection of readings below the intensity threshold

7.5.3.4 *Single movement segmentation*

With this annotation scheme we were finally able to define an alternative method for combining the best aspects of the continuous windowed recognition and isolated recognition approaches. Rather than using two segmentation approaches, we applied a single precisely defined segmentation criteria for both annotation and classification (Figure 41). Based on our candidate gestures of interest, we used the previously established intensity threshold on the yaw or roll axis of the gyroscope readings (*Thresh=100*) as the indicator of a new movement or the end of an existing movement. We used an approach modeled on edge-triggered logic in digital electronic systems. This approach analyzed every two samples to detect transitions from one direction to another.

For example, if a sample without rightward movement (*yaw-*) (defined as before as movement above 100 degrees per second) was followed by sample that did contain a rightward movement, a new group was created. This group persisted until a sample without a

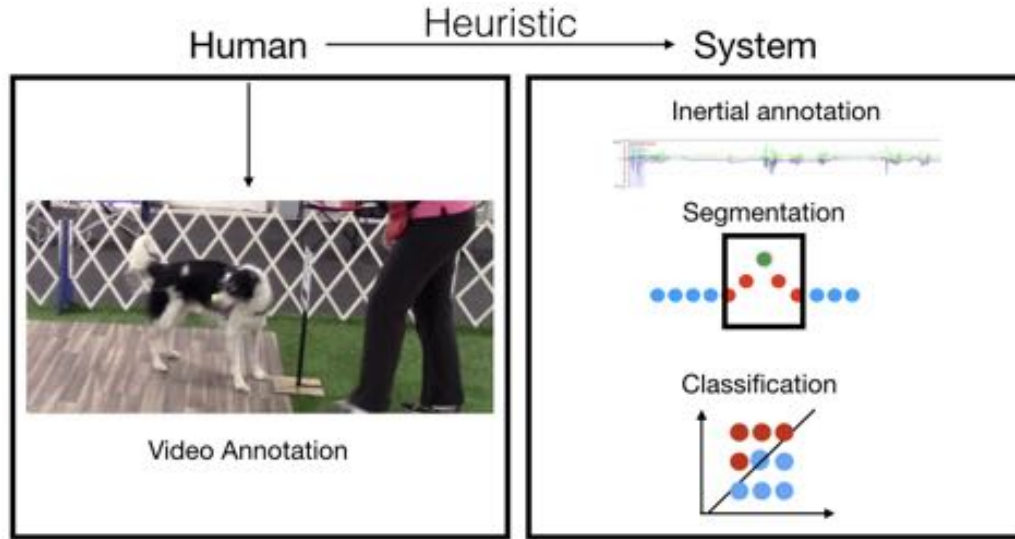


Figure 41. We devised a third approach, where the system applies the annotations programatically through human supervision.

rightward movement was detected. The same applied to other directions on *yaw*, such as *yaw+*, as well as *roll* (*roll+*, *roll-*).

7.5.3.5 Combined movement grouping

Once we had the multi-sample single movements defined, these had to be assembled into our broad movements of interest (candidate gesture movements). We call these *combined movements*, rather than gestures, because not all combined movements will amount to gestures. Unfortunately, unlike the segmentation approach we used for single movements, we learned in the false positive study that we could not use the intensity of single samples as a guide to dictate when a combined movement would begin because *reach* movements had zero-crossings that fell below all intensity thresholds.

In the false positive study, this issue was solved by using a variance threshold on a grouping of $n=40$ sample window. In this study, we would have to consider a concept similar to a fixed window at the combined movement level.

We instead decided to consider all groupings of three multi-sample single movements. These are essentially groups of three segments, which we called *triplets*. We chose three segments because it allowed for single movements to be preceded by non-movements (e.g.,

Space, Right, Space). It also allowed for combining two movements with a potential non-movement, (e.g., *Right Peak, Space, Left Peak*). Finally, in previous experiments we noticed that rotation gestures often involved some brief pauses that caused rotations to result in multiple segments. In this arrangement, a homogeneous movement with pauses (e.g., *Right, Right, Right*), was allowed, and corresponded to a clockwise rotation candidate gesture (*Spin*). In this way, we combined the best aspects of sample-level segmentation, with group-level windowing. That is, we allowed dogs to perform movements of arbitrary lengths, but recognizing that the candidate gestures we considered could be deconstructed into a distinct number of parts.

7.5.3.6 Combined movement annotation

Having created groups of three segments (triplets), it was now necessary to assign them a group class label as well. To recap, we have so far assigned labels to axes, samples, multi-sample segments, and now to combined movements. Our original intention was that the multi-sample segment labels would be enough to assign a group label to the triplet for training. For example, observing the following multi-sample labels in a triple [*Right Peak, Space, Left Peak*] would be enough to label the triplet as a *Right reach*. Unfortunately, this was not the case. With the previous example, the *Space* could have lasted for a long time, such that the *Right Peak* and *Left Peak* were not really part of the same movement.

Instead, it was necessary to examine and constrain the duration of *Space* (sub-threshold activity) between two movements to determine if they belonged to the same candidate gesture. In our case, the reach movements on border collies (BC1, BC2) usually contained gaps of less than a half-second between peaks (Table 31). That meant that the dog's head returned to the forward-looking position rather quickly, rather than performing other behaviors like scratching, etc. For this reason, any delay of longer than half a second or more would lead to the triplets being labeled as not constituting one gesture.

Unfortunately, there was one additional complication. There were cases where the dogs performed the reach movements multiple times in quick succession until they received

Table 31. Annotation parameters that can be set by the human.

Parameter	Value
Intensity threshold	One hundred degrees per second
Maximum time between gestures	half a second
Minimum duration of rotation	one second

feedback. This was not a case we anticipated when the candidate gestures were conceived, but we did not want to penalize this behavior either because it might prove beneficial in some scenarios.

We had previously considered *double reach gesture* in our earlier studies (Chapter 4) and discarded them because they led to detection problems. For example, consider a triplet that contains a *Right Peak* and a *Left Peak* label, but those actions might be parts of two separate movements. It might be the case that the *Right Peak* is the tail end of another movement and the *Left Peak* is the start of a new one. For this reason, labeling this triplet as a *Right Reach* would constitute a false positive.

To address this issue we had to observe the candidate gestures performed and determine a way to make the distinction between two movements of the same gesture and two movements of separate gestures. We finally noticed that the intensity of the first peak (when the head started moving) of gyroscope readings tended to be larger than the intensity of the second peak (when the head returned), so we included this observation into the definition of *reach gestures*. If the gyroscope intensity of the first peak was greater than the intensity of the second peak then we could conclude that the peaks pertained to the same movement and label them as *Right Reach*.

For example, if we look at Figure 42 we can see four peaks (A, B, C, D) corresponding to a dog performing two consecutive *Right Reach* movements. There is not enough information from the inertial data of any single peak for the classifier to infer whether a peak corresponds to the start or the end of a gesture movement, so, both scenarios must be considered. In this example, if the second peak (B) was grouped with the third (C) it would

lead to an incorrect annotation or classification. So we had to ensure A and B would be paired, and C and D would be paired inside a triplet.

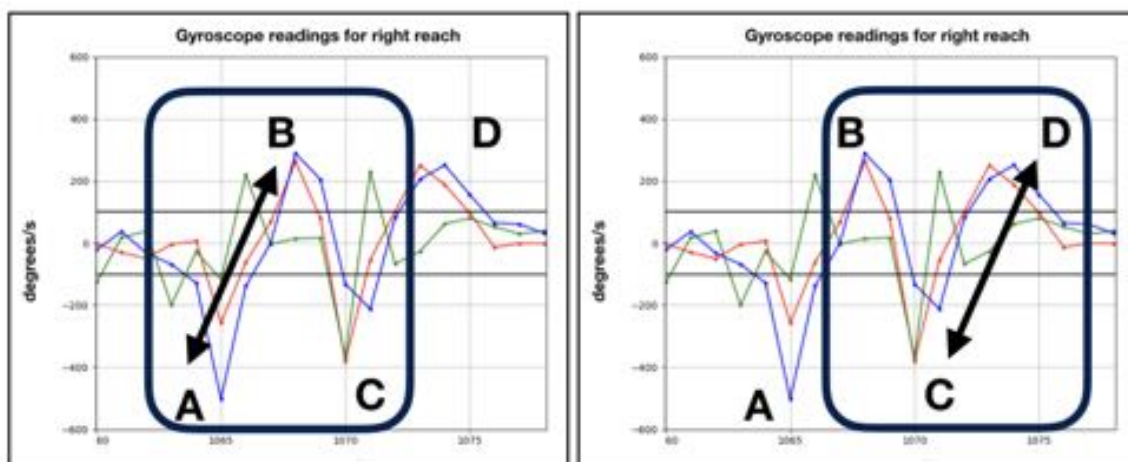


Figure 42. Example of a dog performing one right reach gesture after another.

Finally, we had to address challenges with the candidate gestures consisting of rotations (*Spin*, *Twirl*). Because these candidate gestures consist of movements in only one direction, they heavily depended on duration. Thus, *Right*, *Right*, *Right* might have the same constituent labels as the *Spin* gesture movement, but the duration might be shorter. For our purposes, we defined the duration of a *Spin* (or *Twirl*) to be at least one second.

In summary, labeling a group of three multi-sample segments required analyzing three quantities other than their individual labels. These quantities were, the duration of sub-threshold activity between *reach* gestures, the intensity of peaks in *reach gestures* and the duration of rightward and leftward movement. These observations would later be used to construct a feature vector.

7.5.3.7 Other combined movement annotations

In addition to the gestures we defined as combinations of three multi-sample single movements, there were other possible triplets that were not of interest to our studies, but still formed part of the range of possible dog movements.

For example, triplets that did not meet the definitions established in the preceding section still had to have a combined movement label assigned. One common case were triplets that started with a *Right Peak* of smaller intensity than the subsequent *Left Peak*. This triplet would have qualified as a *Right Reach* except for the small intensity of the first peak, so it was labeled as *Left Reach Small* for diagnostic purposes.

Because we defined originally defined five (5) single movements as building blocks, there were 125 possible combinations of three single movements. When we added the two pitch dependent segments (*Right reach pitch*, *Left reach pitch*) the total increased to seven single movement labels and 343 combined movement labels. In addition, there were different triplet labels for the same three multi-segment labels depending on the intensities of each of the three segments. Also, the position of a movement in the triplets influenced how it was labeled. For example, if a *reach* gesture had a *Space* in the center of the triplet it would be labeled as *reach center*. If, on the other hand, it did not have a *space* in between, it would show up twice in consecutive labels. First it would be visible as [*space*, *Right peak*, *Left peak*], labeled as *right reach early*. Finally, it would have to be visible as [*Right peak*, *Left peak*, *space*], labeled as *right reach late*. We grouped these combinations into a smaller set of 54 categories. Twelve (12) of these categories corresponded to candidate gestures of interest, while the others did not (Table 32).

Some of these categories include very simple combinations. For example, a triplet consisting of *Left* movement accompanied by two *Spaces* might look as follows: [*Space*, *Left*, *Space*]. This triplet would be labeled as *Left center*. We noticed we had to distinguish the position of each movement within the triplet for other non-gesture movements as well, in order to diagnose mis-classifications.

Table 32. Summary of the set of triplet labels defined in this study.

Left Triples	Right Triples	Spaces triples
Left center	Right center	
Left triple	Right triple	
Left peak early	Right peak early	
Left peak center	Right peak center	
Left peak late	Right peak late	
Left peak double	Right peak double	
Left peak double pitch	Right peak double pitch	
Left peak pitch	Right peak pitch	
Left reach early	Right reach early	
Left reach center	Right reach center	
Left reach late	Right reach late	Space
Left reach early pitch	Right reach early pitch	Other
Left reach center pitch	Right reach center pitch	
Left reach late pitch	Right reach late pitch	
Left reach early small	Right reach early small	
Left reach center small	Right reach center small	
Left reach late small	Right reach late small	
Left reach early pitch small	Right reach early pitch small	
Left reach center pitch small	Right reach center pitch small	
Left reach late pitch small	Right reach late pitch small	
Left twirl early	Right spin early	
Left twirl center	Right spin center	
Left twirl late	Right spin late	

7.5.3.8 Annotations for this study

We can now observe the resulting annotations of combined movements for the training session to be used in the present study (Table 33 and Table 34). We can compare these to the video annotations, which we will describe next.

7.5.4 Video annotation

Each session was defined as consisting of at least ten (10) repetitions of a given candidate gesture. This rudimentary count can be conceived as the first attempt at annotating a given session. This count was not sufficient for several reasons. First, the dogs performed other movements than those that were commanded. Secondly, as stated in the Literature Survey (Chapter 2), humans are poor visual inspectors of certain motions present during gestures.

Table 33. Inertial data annotation of gesture triplets for BC1 sessions.

BC1	Inertial Annotation	Video Annotation
Right reach 1	Right reach: 24	Right reach: 24
	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 14	Right reach: 16
	Left reach: 1	Left reach: 0
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Left reach 1	NA	NA
Left reach 2	Right reach: 1	Right reach: 0
	Left reach: 14	Left reach: 14
	Left reach pitch: 1	Left reach pitch: 0
	Spin: 1	Spin: 0
	Twirl: 0	Twirl: 0
Spin 1	Right reach: 1	Right reach: 2
	Left reach: 0	Left reach: 0
	Spin: 9	Spin: 9
	Twirl: 0	Twirl: 0
Spin 2	Right reach: 0	Right reach: 0
	Left reach: 0	Left reach: 0
	Spin: 13	Spin: 15
	Twirl: 0	Twirl: 0
Twirl 1	Right reach: 0	Right reach: 0
	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0
	Twirl: 10	Twirl: 10
Twirl 2	Right reach: 1	Right reach: 0
	Right reach pitch: 1	Right reach pitch: 1
	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0
	Twirl: 10	Twirl: 11

Third, it was not clear whether some movements should count as having performed the desired gesture movement or not. Finally, the human counting provided a tally of the candidate gesture in question, but did not specify where exactly the gesture movement began or ended.

Table 34. Inertial data annotation of gesture triplets for BC2 sessions.

BC2	Inertial Annotation	Video Annotation
Right reach 1	Right reach: 10	Right reach: 10
	Right reach pitch: 4	Right reach pitch: 0
	Left reach: 3	Left reach: 4
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 4	Right reach: 10
	Right reach pitch: 8	Right reach pitch: 0
	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Left reach 1	Right reach: 0	Right reach: 0
	Right reach pitch: 2	Right reach pitch: 0
	Left reach: 8	Left reach: 11
	Left reach pitch: 3	Left reach pitch: 0
	Spin: 0	Spin: 0
Left reach 2	Twirl: 0	Twirl: 0
	NA	NA
	NA	NA
	NA	NA
	NA	NA
Spin 1	Right reach: 1	Right reach: 1
	Right Reach pitch: 1	Right Reach pitch: 0
	Left reach: 3	Left reach: 0
	Left Reach pitch: 1	Left Reach pitch: 0
	Spin: 6	Spin: 6
Spin 2	Twirl: 0	Twirl: 0
	NA	NA
	NA	NA
	NA	NA
	NA	NA
Twirl 1	Right reach: 1	Right reach 0
	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 2	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0
Twirl 2	Twirl: 8	Twirl: 11
	Right reach: 0	Right reach: 0
	Left reach: 3	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0
Twirl 3	Twirl: 8	Twirl: 8
	Right reach: 0	Right reach: 0
	Left reach: 3	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0

7.6 Segmentation and feature selection

As we explained earlier, the segmentation and classification methods were designed to ensure consistency between the outputs of the annotation phase and classification phase, as

such, they are very similar.

7.6.1 Single movement segmentation

We applied the same single precisely defined segmentation criteria for both annotation and classification. Based on our candidate gestures of interest, we used the threshold on the yaw or roll axis of the gyroscope readings ($Thresh=100$ dps) as the indicator of a new movement or the end of an existing movement. We used an approach modeled on edge-triggered logic. This approach analyzed every two samples to detect transitions from one direction to another.

7.6.2 Feature selection

Previously, we described that the human had to provide certain parameters for the inertial annotator to assign sample labels, movement labels, and combined movement labels to each session. The classifier would not have the benefit of knowing these parameters, but should have features to infer them from training data.

In total, we realized that all dog movements can be described in terms of three qualities. They are *intensity*, *direction*, and *duration*. Unfortunately, any one of these three qualities alone is not enough to separate candidate gestures from every day movements.

For example, the *intensity* and *duration* with which a movement is performed varies per dog and even between movements of the same dog. Therefore, the only remaining feature is *direction* or the path taken by dogs in their movement. But the path of a movement without consideration of intensity and duration leads to false positives. So, our challenge was to find a way to combine three sub-optimal features, that individually were not distinguished from everyday movements, into a combination of features that did.

Before describing the set of features we used, it is important to explain how the classifier selection influenced the type of features considered or remained available. For example, we described three components in which we can break down all dog movements. Unfortunately, some classification methods do not naturally preserve this information. For

example, some classifiers such as support vector machines, required features to be regularized and scaled appropriately to ensure that large-scale features do not have an out-sized influence because of their unit size. Unfortunately, some common methods of regularization take away the intensity information, which is vital for classifying dog movements.

Similarly, machine learning classifiers, require equal length feature vectors. Even methods intended for time-series, such as hidden-Markov Models, can be heavily influenced by the length of the segment of interest. As we described earlier, the length of the feature vector itself depends on the segmentation criteria. When we relied on fixed windows, the duration information was also lost because all windows were the same length.

In summary, when attempted a standard approach to feature selection we would lose intensity, and duration, two of the most important features for describing dog movements.

Instead, we began considering a feature set with only a raw collection of gyroscope data of each candidate gesture as a whole. Preserving raw data had the advantage of capturing information of both length and intensity of the segment of interest. Unfortunately, other than using variants of dynamic-time warping as a distance metric, we never succeeded in finding a method of comparing sequences of different lengths.

We then attempted to construct categorical features based on raw data for each individual movement, similar to the scheme we used for annotation labels. For example, a segment containing both x and z positive axis movement might be labeled as *Left Peak* while a segment containing negative x and z axis movement might be labeled *Right Peak*. If only z axis surpassed the defined threshold, the multisegment sample movement would be classified as *Right* or *Left*, respectively.

Unfortunately, this categorical scheme did not preserve any notion of duration of a given segment. As we have mentioned, duration is an important component in full rotation movements and in the small space in between *reach* gestures. We decided to store segment duration as a separate feature for *Right Peak*, *Left Peak*, *Right*, *Left* and their variants.

At this point, the same problems we encountered in the annotation stage persisted in

constructing the features. First, if the dog repeated a *reach* gesture the closing part of the first gesture would be paired with the starting part of a new gesture. In short, *Right Peaks* and *Left Peaks* would be paired with the incorrect complement. Earlier we had already observed that opening peaks were always higher intensity (velocity) than closing ones, and used this information for annotation. Now it became evident, this intensity information had to also be preserved as a feature.

Secondly, there was still no way to codify the acceptable length of readings below the activity threshold inside of a gesture.

The solution involved three steps. First, we stored the maximum intensity value of each segment along each axis (including the sign). Second, we re-defined our groupings of interest to consist of three single segments as in the annotation phase. This scheme allowed for the triplets [*Right Peak, Space, Left Peak*] or [*Right Peak, Left Peak, Space*]. Third and last, we stored the length of the segment as the fourth value (Figure 43). Over time we noticed that we had to apply the same features to our non-movement segments, *Space*, as we did for movement segments. This information included their length.

Finally, we added a positive intensity feature (L2 norm of the maximum values) that was not present in the annotation stage. The reason for this addition was that we were unsure if the classifiers would be able to infer the maximum intensity value of a segment regardless of their numeric sign, and decided to encode it explicitly instead.

[max x intensity, max y intensity, max z intensity, L2,duration]

Figure 43. Example of the feature vector for one multi-sample segment.

7.6.2.1 *Single movement features*

To recap, the features for characterizing single movements consisted of the maximum intensity values of the *x*, *y*, *z* axis gyroscope readings (including sign), the intensity, and the duration of the segment. With this scheme, each single movement, was characterized by

five numbers.

As we later found out, periods of sub-threshold movement or low intensity movements are important components of the candidate gestures so it was important to characterize them (labeled as *Spaces*) using the same five numbers used for high intensity movements.

This is how we concluded with five features for each of our multi-segment single movements.

7.6.2.2 Combined movement features

We can now describe the complete feature vector for combined movements. Combined movements, including candidate gestures, were previously defined as groups of three multi-sample single movements. They are constructed by grouping every three segments, with two segments of overlap.

When combined, the individual features of all three amounted two fifteen numbers to characterize the triplet. Finally, because *Spin* and *Twirl* can be composed of multiple individual segments, we added two features to store the duration of *right* and *left* movements in the candidate gestures as a whole. The final feature set consisted of seventeen (17) numbers, twelve (12) for intensity and five (5) for duration. (Figure 44).

$$\left\{ \begin{array}{l} [\text{max}\textcolor{red}{x}1_{\text{intensity}}, \text{max}\textcolor{green}{y}1_{\text{intensity}}, \text{max}\textcolor{blue}{z}1_{\text{intensity}}, L_2, \text{duration}] \\ [\text{max}\textcolor{red}{x}2_{\text{intensity}}, \text{max}\textcolor{green}{y}2_{\text{intensity}}, \text{max}\textcolor{blue}{z}2_{\text{intensity}}, L_2, \text{duration}] \\ [\text{max}\textcolor{red}{x}3_{\text{intensity}}, \text{max}\textcolor{green}{y}3_{\text{intensity}}, \text{max}\textcolor{blue}{z}3_{\text{intensity}}, L_2, \text{duration}] \\ \text{duration}_{\text{right}}, \text{duration}_{\text{left}} \end{array} \right\}$$

Figure 44. Example of the feature vector for a given triplet.

7.7 Results and classification

For this study we considered a series of different types classifiers. Specifically, we will show results for classifiers based on support vector machines, stochastic gradient descent, k-nearest neighbors, decision trees, and random forests.

We relied on leave-one-session-out validation with BC1 and BC2 sessions as the standard validation technique. We relied on this type of evaluation because it allowed us to use the largest amount of data for training, without compromising the integrity of our results. We did not use standard cross validation evaluation because it could lead to the same data being present in both testing and training set as described earlier [61].

7.7.1 Data sets

The data for this experiment consisted of fourteen (14) sessions from two different subjects (BC1, BC2). They were four (4) *right reach* sessions, two (2) *left reach* sessions, three (3) *spin* sessions, and four (4) *twirl* sessions. They were not chosen in any particular order.

7.7.2 Results of offline evaluation

The first classifier we used was based on a support vector machine (SVM) with radial basis function kernel. At every stage of development the support vector machine seemed to select the most prevalent classes in a session and make all predictions towards those classes (Figure 45).

In general, we believe that support vector machines are not well suited for highly imbalanced classes as we had in this study.

A second alternative was to rely on a stochastic gradient descent (SGD) classifier with a hinge loss function. Although it performed slightly better than the support vector machine, the results were still below our expectations (Figure 46).

We next tried a nearest neighbor classifier (kNN) with $k=2$ neighbors. Although kNN classifiers are considered very simple by current standards, they often have favorable results in ubiquitous computing, although this is perhaps due to the presence of overlapping windows in time series [61]. The results are illustrated in the confusion matrix below (Figure 47).

Finally, we decided to rely on a random-forest based classifier. For this test, the number of estimators was empirically set to ($n_estimators=20$) and maximum depth to ($d=20$). The

Table 35. Full comparison of results for BC1 sessions.

BC1	Inertial Annotation	RF Classifier (LOSO)	Video Annotation
Right reach 1	Right reach: 24	Right reach: 26	Right reach: 24
	Right reach pitch: 1	Right reach pitch: 0	Right reach pitch: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 14	Right reach: 15	Right reach: 16
	Right reach pitch: 1	Right reach pitch: 1	Left reach: 0
	Left reach: 1	Left reach: 0	Spin: 0
	Spin: 0	Spin: 0	Twirl: 0
	Twirl: 0	Twirl: 0	
Left reach 1	NA	NA	NA
Left reach 2	Right reach: 1	Right reach: 0	Right reach: 0
	Left reach: 14	Left reach: 16	Left reach: 14
	Left reach pitch: 1	Left reach pitch: 0	Left reach pitch: 0
	Spin: 1	Spin: 1	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Spin 1	Right reach: 1	Right reach: 1	Right reach: 2
	Right reach pitch: 0	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 9	Spin: 9	Spin: 9
	Twirl: 0	Triple: 0	Twirl: 0
Spin 2	Right reach: 0	Right reach: 0	Right reach: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 13	Spin: 13	Spin: 15
	Twirl: 0	Twirl: 0	Twirl: 0
Twirl 1	Right reach: 0	Right reach: 0	Right reach: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 10	Twirl: 9	Twirl: 10
Twirl 2	Right reach: 1	Right reach: 0	Right reach: 0
	Right reach pitch: 1	Right reach pitch: 0	Right reach pitch: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 10	Twirl: 10	Twirl: 11

In this sense, it is desirable that the annotation system remains separate from the classification or recognition.

7.9 Conclusion

In this chapter we discussed the development of a wearable gesture recognition system. We began by describing procedures for data collection (including dog training) used for system training and testing of a machine learning classifier.

The first step in the data collection process was to consider how to train new subjects to perform candidate gesture movements on environmental cues, rather than cues from their human. During this training, each dog had to perform the candidate gestures at the presence of small-scale traffic signs. We considered a variety of possible cues, but ultimately decided on the traffic signs due to their stability, appropriate height, and distinct shapes.

Table 36. Full comparison of results for BC2 sessions.

BC2	Inertial Annotation	Classification	Video Annotation
Right reach 1	Right reach: 10	Right reach: 9	Right reach: 10
	Right reach pitch: 4	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 3	Left reach: 4	Left reach: 4
	Spin: 0	Spin: 2	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 4	Rich reach: 5	Right reach: 10
	Right reach pitch: 8	Right reach pitch: 3	Right reach pitch: 0
	Left reach: 0	Left reach: 1	Left reach: 0
	Spin: 0	Left reach pitch: 2	Spin: 0
	Twirl: 0	Spin: 0	Twirl: 0
		Twirl: 0	
Left reach 1	Right reach: 0	Right reach: 1	Right reach: 0
	Right reach pitch: 2	Right reach pitch: 0	Right reach pitch: 0
	Left reach: 8	Left reach: 6	Left reach: 11
	Left reach pitch: 3	Left reach pitch: 0	Left reach pitch: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Left reach 2	NA	NA	NA
Spin 1	Right reach: 1	Right reach: 1	Right reach: 1
	Right reach pitch: 1	Right reach pitch: 2	Right reach pitch: 0
	Left reach: 3	Left reach: 4	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 1	Left reach pitch: 0
	Spin: 6	Spin: 6	Spin: 6
	Twirl: 0	Twirl: 0	Twirl: 0
Spin 2	NA	NA	NA
Twirl 1	Right reach: 1	Right reach: 1	Right reach: 0
	Right reach pitch: 1	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 2	Left reach: 3	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 0	Left reach pitch: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 8	Twirl: 7	Twirl: 11
Twirl 2	Right reach: 0	Right reach: 0	Right reach: 0
	Left reach: 3	Left reach: 3	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 8	Twirl: 7	Twirl: 8

At this stage, dogs had to be trained to remember multiple cues (symbols) and perform more than one candidate gesture in a given session. Even though we made great progress with one of our subjects, we ultimately decided to simplify the training of discrimination aspects and focused on the data collection aspects necessary to develop our system.

During the recording sessions each dog performed the candidate gestures differently based on their previous experience, their trainer during the session, and the environment in which the session took place. Furthermore, despite our best efforts we could not standardize the performance of the candidate gesture movement. For example, we initially envisioned

all dogs would pause before and after performing the required movements, but in reality this was not achieved. One reason we could not standardize performance was that the unwanted deviations were initially undetectable to the naked eye and only became evident in post processing. As a result, it was not possible for the trainer to reward only the desired performance during training. These difficulties were instead passed on to the annotation stage.

After obtaining candidate gesture examples from all of our dog participants, we began the process of annotation required for supervised learning.

After devising an annotation system, based on groups of three arbitrary length segments, we constructed a set of 17 features to characterize these movements. We described how all dog movements can be decomposed into intensity, direction and duration and how this information is often lost when using traditional recognition techniques. Finally, the designed feature vectors were passed on to a series of four classifiers to determine which one was best suited for our current application.

Similar to the study in Chapter 6, the random forest classifier achieved the best results. We can now proceed to the final evaluation of our system.

CHAPTER 8

WEARABLE GESTURE SYSTEM EVALUATION

8.1 Introduction

The final evaluation consisted of performing a similar analysis to what we performed in the previous chapter (Chapter 7), but using an unseen reservoir of previously untested data. Furthermore, we decided to perform all tests on a completely subject independent manner.

8.2 Participants

For this study, we used data from three dogs (BC1, BC2, BC3) performing several repetitions of each of our candidate gestures. Data from BC1 and BC2 was the same as in the previous study, but this was the first time we used data from BC3. In particular, we will use BC1 and BC2 sessions as training data to predict the movements of BC3.

Table 37. Participant table. For this study we used data on four subjects.

Participant	Breed	Age (years)	Training	Wearable Exp
L1	Retriever cross	3	assistance	yes
BC1	Border collie	7	assistance	yes
BC2	Border collie	7	assistance	yes
BC3	Border collie	1	none	no

8.3 Inertial and video data annotation

The inertial annotation and video annotation for BC1 and BC2 proceed in the same manner as described for the study in Chapter 7. For the new session in this study, BC3 the video annotation labels are summarized in the following table (Table 8.3).

8.4 Experiment setup

For this final evaluation, the parameters of the classifiers remained the same as in our previous study, but the data sets used for analysis were different.

Table 38. Inertial and video annotation for BC3 sessions.

BC3	Inertial Annotation	Video Annotation
Right reach 1	Right reach: 6	Right reach: 10
	Right reach pitch=2	Right reach pitch: 0
	Left reach: 1	Left reach: 0
	Left reach pitch: 1	Spin: 1
	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 12	Right Reach: 12
	Left reach: 1	Left Reach: 0
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Left reach 1	Right reach: 1	Right Reach: 0
	Right reach pitch: 2	Right reach pitch: 2
	Left reach: 5	Left reach: 5
	Left reach pitch: 2	Left reach pitch: 2
	Spin: 2	Spin: 2
	Twirl: 2	Twirl: 2
Left reach 2	Right reach: 0	Right Reach: 0
	Left reach: 20	Left reach: 12
	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0
Spin 1	Right reach: 2	Right reach: 1
	Left reach: 1	Left reach: 0
	Left reach pitch: 3	Left reach pitch: 0
	Spin: 13	Spin: 11
	Twirl: 0	Twirl: 0
Spin 2	Right reach: 3	Right reach: 0
	Left reach: 1	left reach: 0
	Spin: 13	Spin: 11
	Twirl: 2	Twirl: 0
Twirl 1	Right reach: 2	Right Reach: 0
	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 0	Left reach: 0
	Spin: 2	Spin: 0
	Spin: 6	Twirl: 10
Twirl 2	NA	NA

We relied on the following training data, fourteen (14) sessions from two different subjects (BC1,BC2). They were four (4) *right reach* sessions, two (2) *left reach* sessions, three (3) *spin* sessions, and four (4) *twirl* sessions. The test set consisted of seven (7) sessions of BC3 performing the candidate gestures. These were two (2) *right reach*, two (2) *left reach*, two (2) *spin*, one (1) *twirl*.

8.5 Results

We can now present the results of subject independent testing with BC3. For this purpose we will first show the results of classification of all possible combined movements as we

Table 39. Comparison of labels for BC3 sessions.

BC3	Inertial Annotation	RF Classifier	Video Annotation
Right reach 1	Right reach: 6	Right reach: 6	Right reach: 10
	Right reach pitch: 2	Right reach pitch: 2	Right reach pitch: 0
	Left reach: 1	Left reach: 1	Left reach: 0
	Left reach pitch: 1	Left reach pitch: 0	Left reach pitch: 0
	Spin: 1	Spin: 1	Spin: 1
	Twirl: 0	Twirl: 0	Twirl: 0
Right reach 2	Right reach: 12	Right reach: 12	Right Reach: 12
	Left reach: 1	Left reach: 1	Left Reach: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Left reach 1	Right reach: 1	Right reach: 1	Right Reach: 0
	Right reach pitch: 2	Right reach pitch: 2	Right reach pitch: 2
	Left reach: 5	Left reach: 5	Left reach: 5
	Left reach pitch: 2	Left reach pitch: 2	Left reach pitch: 2
	Spin: 2	Spin: 2	Spin: 2
	Twirl: 2	Twirl: 2	Twirl: 2
Left reach 2	Right reach: 0	Right reach: 0	Right Reach: 0
	Left reach: 20	Left reach: 20	Left reach: 12
	Left reach pitch: 1	Left reach pitch: 1	Left reach pitch: 0
	Spin: 0	Spin: 0	Spin: 0
	Twirl: 0	Twirl: 0	Twirl: 0
Spin 1	Right reach: 2	Right reach: 2	Right reach: 1
	Left reach: 1	Left reach: 1	Left reach: 0
	Left reach pitch: 3	Left reach pitch: 3	Left reach pitch: 0
	Spin: 13	Spin: 13	Spin: 11
	Twirl: 0	Twirl: 0	Twirl: 0
Spin 2	Right reach: 3	Right reach: 3	Right reach: 0
	Left reach: 1	Left reach: 1	left reach: 0
	Spin: 13	Spin: 13	Spin: 11
	Twirl: 2	Twirl: 2	Twirl: 0
Twirl 1	Right reach: 2	Right reach: 2	Right Reach: 0
	Right reach pitch: 1	Right reach pitch: 1	Right reach pitch: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 2	Spin: 2	Spin: 0
	Twirl: 6	Twirl: 6	Twirl: 10
Twirl 2	NA		NA

8.5.3 Breed-independent analysis

Up to this point, we had only conducted analysis with three border collie participants. We has trained L1, but had been unable to perform the gesture movements like the other participants.

Nonetheless, we decided to use the data from BC1 and BC2 to predict movements on a dog of a different breed, in this case Participant L1, a labrador retriever.

Because this dog was not expected to distinguish between *Spin* and *Twirl*, he was allowed to perform either movement in one session. We used this session as a test, and the results can be observed in Table 40.

The results of this test were very encouraging. There was substantial agreement between the video annotations, inertial annotations and the classifier predictions. We discuss

these, and the other results in the discussion section below.

Table 40. Results of the retriever performing spin and twirl movements.

L1	Inertial Annotation	RF classification (LOSO)	Video Annotation
Rotation	Right reach: 0	Right reach: 0	Right reach: 0
	Left reach: 0	Left reach: 0	Left reach: 0
	Spin: 13	Spin: 13	Spin: 13
	Twirl: 2	Twirl: 2	Twirl: 2

8.6 Discussion

In this study we performed our final evaluation of a wearable recognition system.

We can see that based on the data sets used in this experiment, the classifier had more training data (multiple sessions with two dogs) than in our previous study. Because of this, our results improved dramatically. In particular, for the candidate gestures in question, there was almost ideal agreement between the inertial annotation and the classifier predictions.

Nonetheless, like in our previous study, there were some differences between the classifier predictions, inertial annotations and the video annotations. In the previous chapter we outlined three basic reasons.

1. Dog was often out of the video frame.
2. Dog was too fast for human to detect a movement.
3. The human allowed for a movement that was not truly the candidate gesture.

In general, the third reason was the most problematic. We noticed that dogs, especially border collies tried to perform the smallest possible movement to get rewarded. As a result, the behavior in a sense was degrading over time. In addition, if they were not rewarded, rather than performing the movements with maximum effort, they started trying other candidate gestures. While these were often not captured by the human, these movements were recorded inertially, leading to discrepancies.

Similarly, because it was simpler to perform, dogs often tried to perform reaches without having to roll their neck. Instead, they moved their head vertically. This movement was sometimes accepted by the human, because it was almost visually indistinguishable from the roll and yaw movement, but in reality was not the candidate gesture. We noticed that this small difference made those movements more prone to false positives. For example, for BC3, the *Spin 1* session had three (3) instances of apparent *left_reach_pitch* but the human did not notice these. In contrast, in the *Left Reach 1* session, the human noticed four (4) of these variants (2 *left_reach_pitch*, 2 *right_reach_pitch*). Examples of the pitch variants could not be discarded because they were present in a large number of data-sets, and would have to be labeled regardless of whether they were acceptable gestures or not.

It is interesting to note that these gesture variants were more prevalent with border collies. In contrast, the retriever made the movements in a slower, more defined way. This characteristic allowed the human to properly assess whether the candidate gesture was performed or not and hence, there was more agreement between inertial and video annotations.

In conclusion, we believe that as the dogs learn to perform the candidate gestures better, the definitions of the candidate gestures will become more refined as well.

8.7 Future work

Future work can be divided into short term and long term aspects.

The most important aspect remaining is to provide audio feedback to dogs upon performing the first movement of a multi-movement gesture, and once again when the gesture movement is complete. If this feedback is not provided, dogs will try to do the smallest possible movement and see if that produces a reward.

Once we have feedback, for example through an audio beep, we can expand the number of gestures that a dog can learn. For example, with reach gestures, one nuisance was the many different paths that dogs took to travel between two points. This drawback can be

turned into an advantage if each path is defined as a new gesture.

For example, we noticed one dog preferred to reach while moving his head up to the side rather than downwards towards the ribs. Another dog could perform the reach gestures while lying down, but it produced a different path than when standing up. Similarly, we noticed that some dogs performed the *Spin* movement by reaching first and then rotating their body, while others achieved the rotation by moving their head along with their body. All these movements could be redefined as individual gestures, each suited to a particular dog or occupation.

Although our present work was focused on working dogs that work closely with humans (such as guide dogs), we believe it can open communication possibilities for dogs in other occupations. Ideally, we believe this might require customizing the set of gestures based on the occupation and environment.

We would also like to find more ways to numerically evaluate the remaining gesture requirements, like ease of remembering the movement.

Finally, we note that some aspects of this work can be extrapolated to other problems involving time-series. In particular, we believe the notion of classifying the smallest segment of understandable data can be of assistance to practitioners in areas depending heavily on domain experts. This intermediate representation can help inspect and annotate data more quickly and hence lead to better training data and classification. This use of categorical data, combined with edge triggered segmentation, can also allow data to train the individual movements and transitions even if the overall sequence being recognized is not the same. This approach is similar in spirit to hidden Markov models but allows the system designer to directly examine the states and propose any corrections, while in general, Markov models tend to be a black box inaccessible to the system designer.

We also presented a window-less approach to data segmentation which can be useful in other time-series domains such as medical imaging and finance.

8.8 Conclusion

In conclusion, in this dissertation we have developed two types of wearable systems to allow working dogs overcome factors that inhibit dog–human communication.

The first one involved tangible interfaces for direct manipulation while the second one relied on gesture movements to be detected from a collar and used for communication.

In developing the wearable gesture system we discovered a set of often-conflicting requirements a movement must meet to be successful for communication. We made these requirements explicit and examined a series of four gesture movements that could meet them by comparing their similarity against data of everyday movements. We developed a pipeline for annotating and classifying gesture dog movements from inertia data. We annotated the most basic movements and showed how their basic features can be combined to form larger gestures for communication.

The outcome of this research is the development of technologies for a wearable gesture system that allows symbolic communication between working dogs and humans despite differences in perceptual abilities, distance and context.

REFERENCES

- [1] A. P. Rossi and C. Ades, “A dog at the keyboard: using arbitrary signs to communicate requests,” in *Animal Cognition*, vol. 11, pp. 329–338, Springer, 2008.
- [2] M. M. Jackson, G. Valentin, L. Freil, L. Burkeen, C. Zeagler, S. Gilliland, B. Currier, and T. Starner, “Fido: wearable communication interfaces for working dogs,” in *Personal and Ubiquitous Computing*, vol. 19, pp. 155–173, Springer-Verlag, 2015.
- [3] G. Valentin, “Gestural activity recognition for canine-human communication,” in *International Symposium on Wearable Computers: Adjunct Program*, pp. 145–149, ACM, 2014.
- [4] A. Ferworn, A. Sadeghian, K. Barnum, H. Rahnama, H. Pham, C. Erickson, D. Ostrom, and L. Dell’Agnese, “Urban search and rescue with canine augmentation technology,” in *2006 IEEE/SMC International Conference on System of Systems Engineering*, pp. 5–pp, IEEE, 2006.
- [5] W. S. Helton, “Working dogs and the future,” *Canine Ergonomics: The Science of Working Dogs*, p. 325, 2009.
- [6] R. Abrantes, *Dog language*. Dogwise Publishing, 2013.
- [7] C. S. Peirce, *Collected papers of charles sanders peirce*, vol. 5. Harvard University Press, 1974.
- [8] A. H. Freedman, I. Gronau, R. M. Schweizer, D. Ortega-Del Vecchyo, E. Han, P. M. Silva, M. Galaverni, Z. Fan, P. Marx, B. Lorente-Galdos, *et al.*, “Genome sequencing highlights the dynamic early history of dogs,” *PLoS Genet*, vol. 10, no. 1, 2014.
- [9] C. J. Pfaffenberger, J. Scott, J. Fuller, B. Ginsburg, S. Biefelt, *et al.*, *Guide dogs for the blind: their selection, development, and training*. Elsevier Scientific Publishing Company., 1976.
- [10] K. G. Furton and L. J. Myers, “Talanta,” *The scientific foundation and efficacy of the use of canines as chemical detectors for explosives*, vol. 54, pp. 487–500, 2001.
- [11] J. W. Pilley and H. Hinzmann, *Chaser: Unlocking the genius of the dog who knows a thousand words*. Houghton Mifflin Harcourt, 2013.
- [12] J. Savage, R. Sanchez-Guzman, W. Mayol-Cuevas, L. Arce, A. Hernandez, L. Brier, F. Martinez, A. Velazquez, and G. Lopez, “Animal-machine interfaces,” in *Wearable Computers, The Fourth International Symposium on*, pp. 191–192, IEEE, 2000.

- [13] W. R. Britt, J. Miller, P. Waggoner, D. M. Bevly, and J. A. Hamilton Jr, “An embedded system for real-time navigation and remote command of a trained canine,” in *Personal and Ubiquitous Computing*, vol. 15, pp. 61–74, Springer, 2011.
- [14] G. Lemasson, S. Pesty, and D. Duhaut, “Increasing communication between a man and a dog,” in *Cognitive Infocommunications (CogInfoCom), 2013 IEEE 4th International Conference on*, pp. 145–148, IEEE, 2013.
- [15] B. I. Resner, *Rover at Home: Computer mediated remote interaction between humans and dogs*. PhD thesis, Massachusetts Institute of Technology, 2001.
- [16] A. Weilenmann and O. Juhlin, “Understanding people and animals: the use of a positioning system in ordinary human-canine interaction,” in *SIGCHI conference on human factors in computing systems*, pp. 2631–2640, ACM, 2011.
- [17] S. Bulanda, *Ready! 2nd Edition The Training of the Search and Rescue Dog*. Kennel Club Books, 2010.
- [18] A. Kendon, *Gesture: Visible action as utterance*. Cambridge University Press, 2004.
- [19] A. Glinsky, *Theremin: ether music and espionage*. University of Illinois Press, 2000.
- [20] A. F. Bobick and A. D. Wilson, “A state-based approach to the representation and recognition of gesture,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 12, pp. 1325–1337, 1997.
- [21] J. Mankoff, S. E. Hudson, and G. D. Abowd, “Providing integrated toolkit-level support for ambiguity in recognition-based interfaces,” in *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pp. 368–375, ACM, 2000.
- [22] W.-C. Bang, W. Chang, K.-H. Kang, E.-S. Choi, A. Potanin, and D.-Y. Kim, “Self-contained spatial input device for wearable computers,” in *null*, p. 26, IEEE, 2003.
- [23] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill, “A hand gesture interface device,” in *ACM SIGCHI Bulletin*, vol. 18, pp. 189–192, ACM, 1987.
- [24] T. E. Starner, “Visual recognition of american sign language using hidden markov models.,” tech. rep., MASSACHUSETTS INST OF TECH CAMBRIDGE DEPT OF BRAIN AND COGNITIVE SCIENCES, 1995.
- [25] C. Cedras and M. Shah, “Motion-based recognition a survey,” *Image and Vision Computing*, vol. 13, no. 2, pp. 129–155, 1995.
- [26] D. M. Gavrila, “The visual analysis of human movement: A survey,” *Computer vision and image understanding*, vol. 73, no. 1, pp. 82–98, 1999.
- [27] T. B. Moeslund, A. Hilton, and V. Krüger, “A survey of advances in vision-based human motion capture and analysis,” *Computer vision and image understanding*, vol. 104, no. 2, pp. 90–126, 2006.

- [28] R. Poppe, "A survey on vision-based human action recognition," *Image and vision computing*, vol. 28, no. 6, pp. 976–990, 2010.
- [29] A. F. Bobick, "Movement, activity and action: the role of knowledge in the perception of motion," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 352, no. 1358, pp. 1257–1265, 1997.
- [30] J. J. LaViola, "A survey of hand posture and gesture recognition techniques and technology," tech. rep., Technical report, 1999.
- [31] S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 3, pp. 311–324, 2007.
- [32] R. Vadehra, "Gesture recognition technology," *International Journal of Computer Applications*, 2007.
- [33] J. F. Bartlett, "Rock'n'scroll is here to stay [user interface]," *IEEE Computer Graphics and Applications*, vol. 20, no. 3, pp. 40–45, 2000.
- [34] A. Y. Benbasat and J. A. Paradiso, "An inertial measurement framework for gesture recognition and applications," in *International Gesture Workshop*, pp. 9–20, Springer, 2001.
- [35] T. Schlömer, B. Poppinga, N. Henze, and S. Boll, "Gesture recognition with a wii controller," in *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pp. 11–14, ACM, 2008.
- [36] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uwave: Accelerometer-based personalized gesture recognition and its applications," *Pervasive and Mobile Computing*, vol. 5, no. 6, pp. 657–675, 2009.
- [37] D. A. Norman and J. Nielsen, "Gestural interfaces: a step backward in usability," *interactions*, vol. 17, no. 5, pp. 46–49, 2010.
- [38] C. R. Stevens and D. E. Reamer, "Method and apparatus for activating switches in response to different acoustic signals," Feb. 20 1996. US Patent 5,493,618.
- [39] D. Ashbrook and T. Starner, "Magic: a motion gesture design tool," in *SIGCHI Conference on Human Factors in Computing Systems*, pp. 2159–2168, ACM, 2010.
- [40] H.-H. Nagel, "From image sequences towards conceptual descriptions," *Image and vision computing*, vol. 6, no. 2, pp. 59–74, 1988.
- [41] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 46, no. 3, p. 33, 2014.
- [42] L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in *International Conference on Pervasive Computing*, pp. 1–17, Springer, 2004.

- [43] C. Ribeiro, A. Ferworn, M. Denko, and J. Tran, "Canine pose estimation: A computing for public safety solution," in *Canadian Conference on Computer and Robot Vision*, pp. 37–44, IEEE, 2009.
- [44] C. Ribeiro, A. Ferworn, M. Denko, J. Tran, and C. Mawson, "Wireless estimation of canine pose for search and rescue," in *System of Systems Engineering. IEEE International Conference on*, pp. 1–6, IEEE, 2008.
- [45] R. Brugarolas, D. Roberts, B. Sherman, and A. Bozkurt, "Posture estimation for a canine machine interface based training system," in *Annual International Conference on Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2012.
- [46] R. Brugarolas, D. Roberts, B. Sherman, and A. Bozkurt, "Machine learning based posture estimation for a wireless canine machine interface," in *Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems (BioWireless)*, pp. 10–12, IEEE, 2013.
- [47] R. Brugarolas, R. T. Loftin, P. Yang, D. L. Roberts, B. Sherman, and A. Bozkurt, "Behavior recognition based on machine learning algorithms for a wireless canine machine interface," in *International Conference on Body Sensor Networks (BSN)*, pp. 1–5, IEEE, 2013.
- [48] R. Morrison, V. Penpraze, R. Greening, T. Underwood, J. J. Reilly, and P. S. Yam, "Correlates of objectively measured physical activity in dogs," *The Veterinary Journal*, vol. 199, no. 2, pp. 263–267, 2014.
- [49] B. D. Hansen, B. D. X. Lascelles, B. W. Keene, A. K. Adams, and A. E. Thomson, "Evaluation of an accelerometer for at-home monitoring of spontaneous activity in dogs," *American journal of veterinary research*, vol. 68, no. 5, pp. 468–475, 2007.
- [50] J. M. Yashari, C. G. Duncan, and F. M. Duerr, "Evaluation of a novel canine activity monitor for at-home physical activity analysis," *BMC veterinary research*, vol. 11, no. 1, p. 1, 2015.
- [51] C. Dow, K. E. Michel, M. Love, and D. C. Brown, "Evaluation of optimal sampling interval for activity monitoring in companion dogs," *American journal of veterinary research*, vol. 70, no. 4, pp. 444–448, 2009.
- [52] D. C. Brown, R. C. Boston, and J. T. Farrar, "Use of an activity monitor to detect response to treatment in dogs with osteoarthritis," *Journal of the American Veterinary Medical Association*, vol. 237, no. 1, pp. 66–70, 2010.
- [53] C. Ladha, N. Hammerla, E. Hughes, P. Olivier, and T. Plötz, "Dog's life: wearable activity recognition for dogs," in *Proc. of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pp. 415–418, ACM, 2013.
- [54] M. P. Groover, *Fundamentals of modern manufacturing: materials processes, and systems*. John Wiley & Sons, 2007.

- [55] A. Geitgey, “Machine learning is fun!,” 2017. Version 1.
- [56] T. Plötz, N. Y. Hammerla, and P. Olivier, “Feature learning for activity recognition in ubiquitous computing,” in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, p. 1729, 2011.
- [57] H. Junker, O. Amft, P. Lukowicz, and G. Tröster, “Gesture spotting with body-worn inertial sensors to detect user activities,” *Pattern Recognition*, vol. 41, no. 6, pp. 2010–2024, 2008.
- [58] G. Berns, *How Dogs Love Us: a neuroscientist and his dog decode the canine brain*. Scribe Publications, 2014.
- [59] R. Coppinger, L. Coppinger, and E. Skillings, “Observations on assistance dog training and use,” *Journal of Applied Animal Welfare Science*, vol. 1, no. 2, pp. 133–144, 1998.
- [60] G. Valentin, J. Alcaininho, L. Freil, C. Zeagler, M. Jackson, and T. Starner, “Canine reachability of snout-based wearable inputs,” in *Proceedings of the ACM International Symposium on Wearable Computers*, pp. 141–142, ACM, 2014.
- [61] N. Y. Hammerla and T. Plötz, “Let’s (not) stick together: pairwise similarity biases cross-validation in activity recognition,” in *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pp. 1041–1051, ACM, 2015.
- [62] I. Axivity, “WAX9 application developer’s guide,” 2014. Version 2.
- [63] M. B. Alexander, T. Friend, and L. Haug, “Obedience training effects on search dog performance,” in *Applied Animal Behaviour Science*, vol. 132, pp. 152–159, Elsevier, 2011.
- [64] P. E. Miller and C. J. Murphy, “Vision in dogs,” in *Journal-American Veterinary Medical Association*, vol. 207, pp. 1623–1634, AVMA, 1995.
- [65] J. Neitz, T. Geist, and G. H. Jacobs, “Color vision in the dog,” in *Visual neuroscience*, vol. 3, pp. 119–125, Cambridge Univ Press, 1989.
- [66] E. Hiby, N. Rooney, and J. Bradshaw, “Dog training methods: their use, effectiveness and interaction with behaviour and welfare,” in *Animal Welfare*, vol. 13, pp. 63–70, Universities Federation for Animal Welfare, 2004.
- [67] M. Bentosela, G. Barrera, A. Jakovcevic, A. M. Elgier, and A. E. Mustaca, “Effect of reinforcement, reinforcer omission and extinction on a communicative response in domestic dogs,” in *Behavioural processes*, vol. 78, pp. 464–469, Elsevier, 2008.
- [68] K. Pryor, *Reaching the animal mind: clicker training and what it teaches us about all animals*. NY: Simon and Schuster, 2009.

- [69] K. Yonezawa, T. Miyaki, and J. Rekimoto, “Cat at log: sensing device attachable to pet cats for supporting human-pet interaction,” in *International Conference on Advances in Computer Entertainment Technology*, pp. 149–156, ACM, 2009.
- [70] B. Velichkovsky, A. Sprenger, and P. Unema, “Towards gaze-mediated interaction: Collecting solutions of the midas touch problem,” in *Human-Computer Interaction INTERACT97*, pp. 509–516, Springer, 1997.
- [71] G. Valentin, J. Alcaidinho, A. Howard, M. M. Jackson, and T. Starner, “Towards a canine-human communication system based on head gestures,” in *12th International Conference on Advances in Computer Entertainment Technology*, vol. 1, (Iskandar), pp. 1–6, ACI, ACM, 2015.