Using Contextual Information to Improve Hidden Markov Model Recognition of Wrist Motions During Eating Activities

---

A Dissertation
Presented to
the Graduate School of
Clemson University

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

---

by
Yiru Shen
December 2018

Accepted by:
Dr. Adam Hoover, Committee Chair
Dr. Jon Calhoun
Dr. Eric Muth
Dr. Yongqiang Wang

Abstract

This dissertation considers the problem of recognizing wrist motions during eating. The wrist motion is tracked using accelerometer and gyroscope sensors worn in a watch-like device. The goal is to recognize a set of gestures that commonly occur during eating, such as taking a bite of food or consuming a drink of liquid. The wrist motion during a bite may consist of picking up a morsel of food, moving it the mouth for intake, and returning the hand to a rest position. Hidden Markov models (HMMs) are used to recognize these motions. An HMM is a doubly stochastic process where one set of stochastic processes generates the observables, in this case the sensor readings, and is controlled by another set of stochastic processes that is not observable, in this case the eating gestures. A benefit of an HMM is that it can encode the temporal structure of the signal, in this case the expected subsequence of motions comprising a gesture.

The ideas pursued in this dissertation are motivated by methods used to improve the capability of HMMs to recognize speech. For example, it is challenging to build a generic HMM to capture all the varieties of accents and dialects of speakers. People in different regions may speak the same language with variations in pronunciation, vocabularies, and grammars. Building HMMs for each dialect group can improve the robustness of the system under speech variations. This dissertation attempts the similar analysis of wrist motion during eating activities. Similar to dialects and accents in speech, we propose that the demographics (gender, age, ethnicity), utensil being used, or types of foods being eaten, may cause variations in the wrist motions while eating. Several variations on this concept are explored and compared to baseline recognition accuracies.

In Chapter 2, work is first described to establish a baseline accuracy of a non-HMM method. The method uses a simple pattern matching algorithm that only detects one type of gesture (called "bites" but includes any food or liquid intake). The method was tested on 276 people eating a meal in a cafeteria and was evaluated on 24,088 bites. It achieved 75% sensitivity and 89% positive

predictive value. Chapter 3 describes a larger vocabulary of eating actions using segment-based labeling. The set of gestures include taking a bite of food (bite), sipping a drink of liquid (drink), manipulating food for preparation of intake (utensiling), and not moving (rest). All other activities such as using a napkin or gesturing while talking are grouped into a non-eating category (other). The lexicography was tested by labeling segments of wrist motion according to the gesture set. A total of 18 human raters labeled the same data used described above. Inter-rater reliability was 92.5% demonstrating reasonable consistency of gesture definitions. Chapter 4 describes work that explores the complexity of HMMs and the amount of training data needed to adequately capture the motion variability across the large data set. Results found that HMMs needed a complexity of 13 states and 5 Gaussians to reach a plateau in accuracy, signifying that a minimum of 65 samples per gesture type are needed. Results also found that 500 training samples per gesture type were needed to identify the point of diminishing returns in recognition accuracy. Overall, it achieved 85.2% all gestures accuracy for HMM-S that models a single gesture as a sequence of sub-gestures. It also achieved 89.5% all gestures accuracy for HMM-1, where a sequence of one previous gesture was studied as context. Chapter 5 describes work that investigates contextual variables to recognize gestures using top-down and bottom-up approaches. Specifically, we consider if foreknowledge of the demographics (gender, age, hand used, ethnicity, BMI), meal level variables (utensil used for eating, food consumed), language variables (variations of bite, utensiling and other), and clustering based method can improve recognition accuracy. We investigate this hypothesis by building HMMs trained for each of these contextual variables, and compare their accuracy against the simple non-HMM algorithm and HMM-S. Results show that the highest accuracy of all gestures and intake gestures in contextual HMMs is 86.4% and 91.7%, improved by 1.2% and 6.7% over HMM-S, respectively. We also investigate the contextual variables along with one gesture history. It achieved all gestures accuracy up to 88.9% and intake gestures accuracy up to 93.0%, with 0.6% decreased for all gestures accuracy and 1.5% intake gestures accuracy improved over HMM-1.

Acknowledgments

I would like to sincerely thank my advisor, Dr. Adam Hoover, for all the guidance, patience and advice throughout my journey at Clemson University. It is not easy to pursue a doctoral degree in the past 5 and a half years, challenge is not only in research but also in daily life. Mentored by Dr. Hoover, I am fortunate to realize that one knows something as an undergraduate, one knows a little more as a master student, and one knows nothing as a PhD student! He helped me build up my critical thinking in research: what is the problem, why is it a problem, what have others done and what have we done and also in communication with people: always be kind and patient. All that I have accomplished in this journey would not have been possible without his support. I would also like to thank my committee members, Dr. Jon Calhoun, Dr. Eric Muth and Dr. Yongqiang Wang for their time and valuable feedbacks on my dissertation. Also, I would like to thank Dr. Daniel Noneaker for the financial support during my time in Clemson. In addition, I would like to express my deepest gratitude to my parents, for encouraging me to be a better person, be my inspiration and always stand by me, providing care, patience and endless love. Thanks also belong to my dear friends and colleagues, Jingxuan Sun, Sufeng Niu, Raul Ramos-Garcia, Chen Feng, Dong Tian, among many others, who helped to find a way to lift me up.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This work considers the problem of recognizing eating activities by tracking wrist motion. The wrist motion is recorded by a watch-like device worn by participants while eating an unscripted meal. Prior research has been investigating methods based upon wrist motion tracking [25, 26, 27, 97, 106]. Previous work [25, 26, 27] describes a pattern of motion indicative of hand-to-mouth gestures and an algorithm to detect and count their occurrences, which we call bite counting. Previous work [97, 106] describes methods using hidden Markov models (HMMs) to detect five different types of gestures (food bite, drink bite, utensiling, rest, and other), and an algorithm that improves their recognition through gesture-to-gesture sequential modeling.

In this dissertation, the proposed method is evaluated against two benchmarks. The first one is a simple bite counting algorithm of only detecting a single gesture "bite" that includes all intakes of food or liquid [107]. The second one recognizes five different gesture types but models all types of each gesture using a single generic HMM: HMM-S. However, a large variety of motions exists in gestures and a generic model is not capable to recognize all the patterns. For example, the wrist motion of taking a bite with fork is different from taking a bite with both hands. Under this circumstance, this work studies the contextual variables to vary the gesture vocabularies and reduce motion variations with the goal to improve recognition accuracy. First, a vocabulary of actions is proposed to quantify gestural behaviors during eating based on discernible intent. Second, the gesture information captured by each contextual variable is studied by HMMs. Two evaluation metrics are considered here: general accuracy that evaluates the performance of recognizing all the eating gestures, and intake accuracy that evaluate the performance of recognizing the gestures

specific to energy intake.

The following sections provide background information regarding the fields of study being discussed in this work. Section 1.1 describes some concepts from speech recognition that we apply to the problem of recognizing eating gestures. Section 1.2 describes the health problem that motivates this work. Section 1.3 introduces the mobile health and their applications in various fields. Section 1.4 discusses wearable sensors used for dietary monitoring. Section 1.5 gives the information of Micro-Electro-Mechanical Systems (MEMS) sensors and the sensors used in this work. Section 1.6 explains the theory of hidden Markov models.

1.1 Language of Eating

This section introduces the background in speech recognition and the similar concepts we used in eating gesture recognition.

1.1.1 Background of speech recognition

Figure 1.1 shows an example of a speech signal of three words from the TIDIGITS corpus [66]. To determine the word type, each segment of signal can be recognized independently, which is known as isolated word recognition. In this model, the basic speech unit is the word, where the goal is to recognize a single spoken word. The speech is converted from the analog signal captured by a microphone to a digital signal by sampling over a second. Each value is quantized to 16 bits. Each signal segment is divided into frames/windows with the same duration, usually 20 to 30 ms (320 to 480 samples). Features are then extracted into a multiple-element feature vector per frame.

To recognize isolated words, one HMM is built for one word and used to learn the sequential information. Frames with sliding window are used to extract feature vectors inside each frame. These feature vectors are used to train HMMs. The amount of HMMs depends on the number of words in the target. During test, feature vectors of each word are passed into these HMMs and the one with the max score determines word type. Figure 1.2 illustrates the process. Details of training HMMs is introduced in Section 1.6.

Other applications include connected word recognition, where each word is segmented from a continuous speech signal and then recognized by building HMMs. During segmenting, silence signals, as the pauses between speech, are helpful to obtain voiced part. Two approaches have been

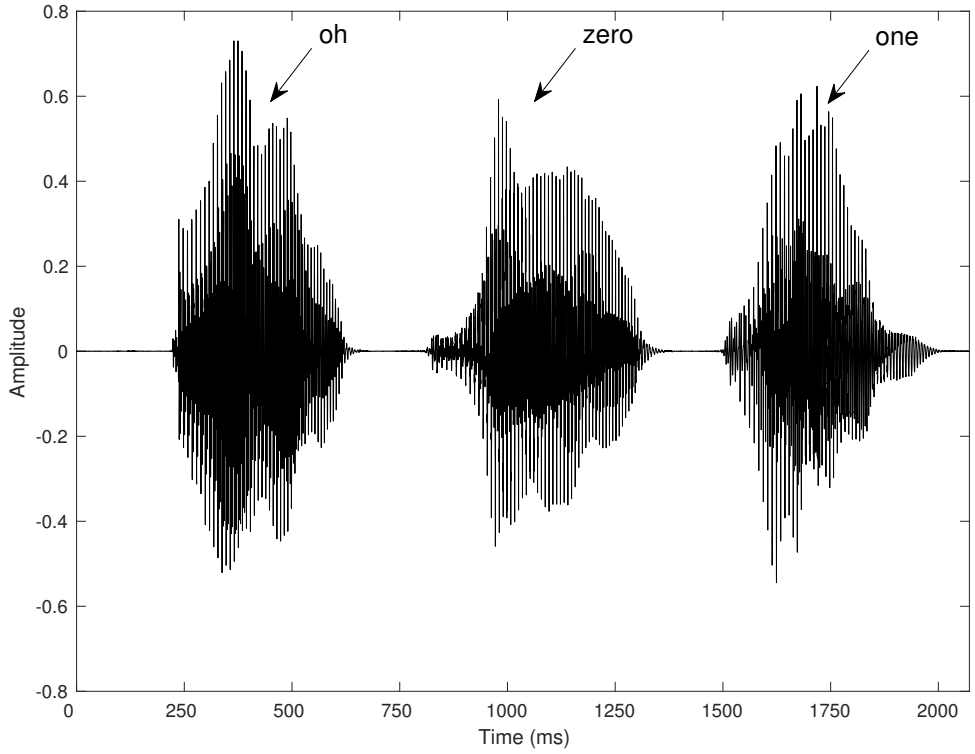


Figure 1.1: Speech signal of word "oh", "zero", "one" from TIDIGITS corpus [66] sampled using 8 kHz sampling frequency.

widely taken for silence removal: Short Time Energy (STE) and Zeros Crossing Rate (ZCR) [11, 19]. STE detects silence signal by the fact that energy in voiced sample is greater than silence/unvoiced sample, while ZCR detects it by the amount of zero crossings within a portion of speech. Once silence signals are removed, segments containing voiced information are passed into the process as shown in Figure 1.2.

One problem in speech recognition is handling multiple dialects and accents. People in different groups may speak the same language with variations in pronunciation, vocabularies, and grammars. It is challenging in automatic speech recognition because the system must be able to recognize all the potential variations. Thus being able to accurately classify dialects and accents makes the system adapt the pronunciation, acoustic, and language features accordingly and can improve the system robustness. One approach works on building dialect-specific models for each dialect group and recognize the dialects with the max similarity score [48, 59]. Other approaches

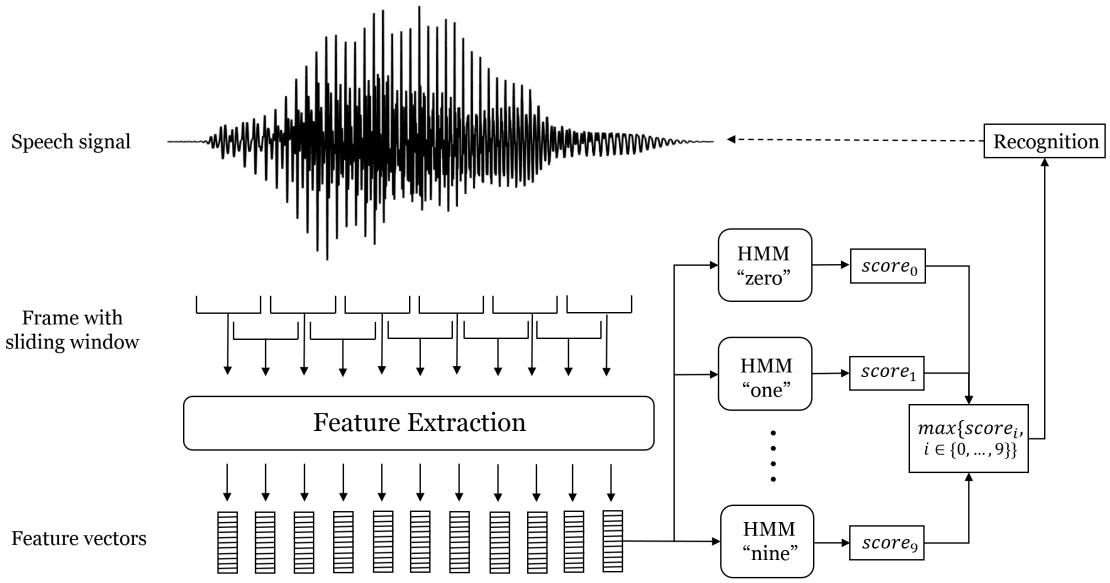


Figure 1.2: The process of extracting feature sequences of one word and recognizing the word from digit 0-9.

include language modeling to capture statistics of phonotactic distributions [120], learning specific lexical and acoustic features [17, 55], or adapting models trained on generic speech using Maximum Likelihood Linear Regression to each test speaker [14, 47, 119, 125]. In this dissertation, we try the same ideas of building models of different contextual variables to reduce motion variaties for recognizing eating gestures.

1.1.2 Eating gesture recognition

In this section, we introduce our study on eating gesture recognition and its similarity and difference with speech recognition. Both speech and the motion signals are captured over time and contain temporal structures. Different from speech that is inherently a one-dimensional continuous signal across time, our motion signals are tracked by a wrist-worn device with 3-axis gyroscopes and 3-axis accelerometers MEMS sensors and hence contain more information. For example, a 3-axis accelerometer sensor captures the instant orientation of the wrist and can indicate the poses while eating. The gyroscopes capture rotational velocity of the wrist for yaw, pitch and roll motion. Previous work in our group has discovered that the wrist of a person undergoes a characteristic rolling motion that is indicative of the person taking a bite of food [46]. This characteristic roll

motion can help differentiate wrist motions from different activities. For example, the roll motion of moving food around a plate is different from non-eating-related activities and can help detect the event of taking a bite of food.

Second, the vocabulary in speech and wrist motion is different. Spoken words can be broken into eight categories of nouns, determiners, pronouns, verbs, adjectives, adverbs, prepositions, and conjunctions [1]. In general, wrist motion words describe actions while eating. For example, word "bite" describes a series of movements when food is put through mouth for consumption. On the other hand, vocabulary size differs. The vocabulary size of the spoken words of a typical young adult is 10,000-50,000 [1]. In speech recognition systems, the vocabulary size depends on the applications. Generally, small, medium and large vocabulary size are the order of 100, 1000 and (over) 5000 words [39]. For example, a small vocabulary model can recognize only ten digits. In wrist motion recognition, our target is to recognize actions while eating, thus the typical word set is 5-20, which is much smaller than speech vocabulary. Specifically, five gestures are defined in this work based on discernible intent: taking a bite of food (bite), sipping a drink of liquid (drink), manipulating food for preparation of intake (utensiling), and not moving (rest). All other activities such as using a napkin or gesturing while talking are grouped into a non-eating category (other).

Third, the "dialect" in speech and wrist motions are different. Dialect in speech refers to the same word with varieties in pronunciation, vocabularies, and grammars, while the "dialect" in our work refers to the varieties of wrist motions of accomplishing the same action. For example, cutting or stirring food, or dipping food into sauces are all considered as the word utensiling, although different motion patterns exist.

Figure 1.3 shows the sampled sensor data and the gesture sequence. Details of the gesture vocabulary will be discussed in Chapter 3. Similar to isolated word recognition as outlined in Figure 1.2, after obtaining the gesture segments, we extract features from six sensors and use them to build HMMs for gesture recognition.

1.2 Motivating health problem

The motivating health problem of this work is obesity. Overweight and obesity are defined as abnormal or excessive fat accumulation that may impair health. Body mass index (BMI) is a simple index of weight-for-height that is commonly used to classify overweight and obesity in adults. BMI

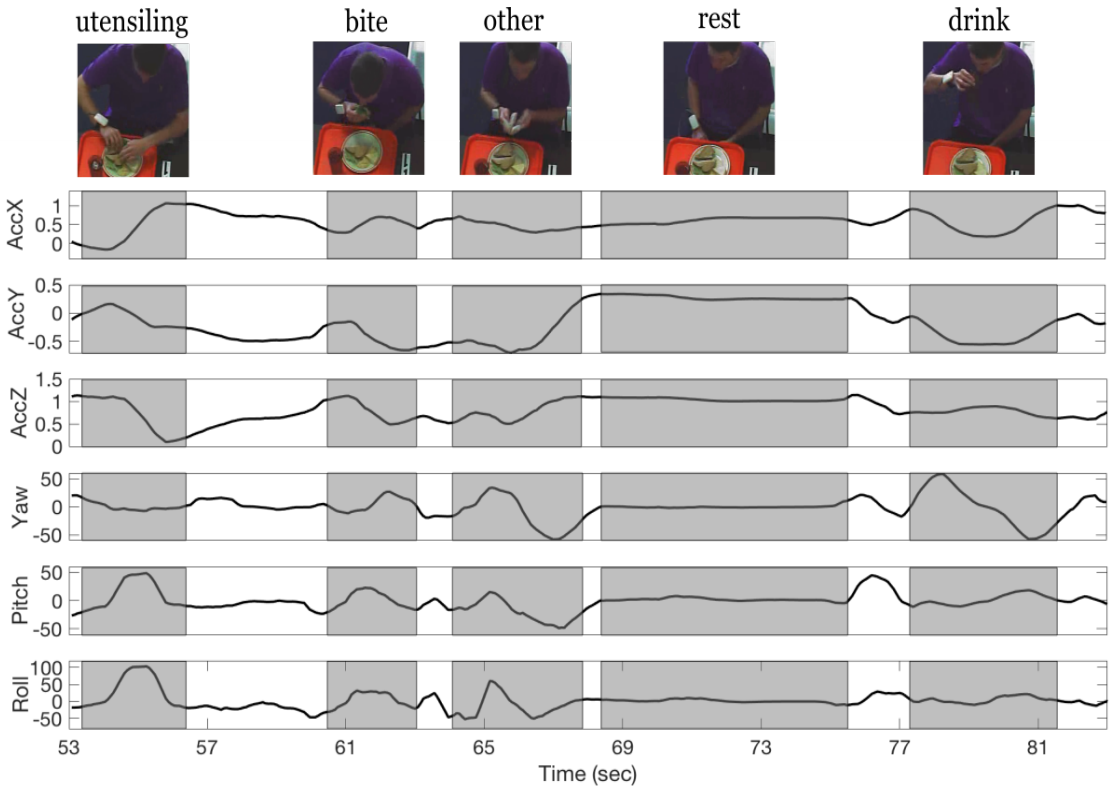


Figure 1.3: Sensor data of eating gesture sequence. From top to bottom: acceleration of AccX, AccY, AccZ in unit gravity (g) and rotational velocity measured by gyroscope of yaw, pitch and roll, in unit degree per second (deg/sec). From left to right: utensiling, bite, other, rest and drink. Shaded regions indicate the correponding gestures. The unit of AccX, AccY and AccZ is gravity (g) and the unit of yaw, pitch and roll is degree per second.

is defined as a person's weight in kilograms divided by the square of his height in meters (kg/m^2). The World Health Organization (WHO) defines overweight as a BMI≥25, obese as a BMI≥30, and morbidly obese as a BMI≥40. More than half of the world population is overweight (39%) or obese (13%). In the U.S., 17% of children and more than 30% of adults are considered obese, with 2.8% of males and 6.9% of females are extremely obese (body mass index $\geq$ 40) [83]. The prevalence of overweight and obesity has increased markedly in the last 2 decades in the United States [43]. Obesity is associated with increased risks for several diseases, including cardiovascular diseases (mainly heart disease and stroke), diabetes, musculoskeletal disorders (especially osteoarthritis a highly disabling degenerative disease of the joints) and some cancers (including endometrial, breast, ovarian, prostate, liver, gallbladder, kidney, and colon) [72]. The increased prevalence of obesity is responsible for almost $40 billion of increased medical spending through 2006, including $7 billion in Medicare prescription drug costs [34]. Recent studies have concluded that if obesity were to remain at 2010 levels, the combined savings in medical expenditures over the next 2 decades would be $549.5 billion, with a 33% estimated increase in obesity prevalence and a 130% increase in severe obesity prevalence over the next 2 decades [33].

One problem in obesity is to measure energy intake over time. It is challenging and time consuming to record the daily consumption of food and beverage. Conventional methods for measuring energy intake include manual entry of self-reported intake into food diaries and 24-hour recalls [44, 103]. However, these methods are prone to under-reporting and under-estimation and are tedious to use resulting in non-compliance over the long term. Recently, body worn sensors have been investigated to automatically measure energy intake by tracking eating related motions. For example, various types of sensors have been worn on different body locations to detect eating related activities such as chewing and swallowing to estimate intake calories [102]. This dissertation investigates recognizing eating related gestures by using wearable sensors. A future goal would be to convert detected gestures into an estimate of energy intake.

1.3 Mobile health

Mobile health, referred to as mHealth, offers the ability to improve a subject's health by monitoring their status, recognizing behaviors, diagnosing medical conditions, and providing interventions if necessary through the use of wireless portable devices [3]. For example, people can use

applications that run on mobile phones and sensors that track vital signs and health activities for the track of blood glucose, carbohydrate, insulin doses and activity [29]. Other work reports several types of mobile medical applications [13]. One is used as an extension of a device such as a remote display of data from a bedside monitor. A second group are the mobile applications that transfer mobile platform into a medical device by using display screens, or sensors to those of currently regulated medical devices. A third group are the mobile applications that allow the user to input patient-specific information and through the use of formulae or processing algorithms, output a patient-specific result, diagnosis, or treatment recommendation to be used in clinical practice or to assist in making clinical decisions.

Recently, mHealth has been researched for self-monitoring of weight control [116]. Smartphones and smartwatches can provide an opportunity for real-time feedbacks on weight-related behaviors. For example, mobile phones were used to weekly deliver short message service (SMS) of diet and exercise information to each user. Result showed that this service is an effective method of behavior modification in weight control [121]. On the other hand, custom applications can be built into the smartphone to allow users to self-monitor caloric balance in real time [51, 69].

1.4 Wearable Sensors for Dietary Monitoring

Wearable sensors, as the typical mHealth tools, have been widely studied in the field of dietary monitoring to automatically measure energy intake. Several sensing approaches have been studied, including sensing location and modality to detect eating instances. Besides, several positions on the human body can be instrumented to detect activities associated with eating. A general review for wearable sensor based systems can be found for automatic dietary monitoring [94]. In this section, we discuss some works of wearable sensors used for dietary monitoring based on the sensing modality: acoustic sensing system, camera-based sensing system, motion-based sensing system and multimodal sensing system.

1.4.1 Acoustic Sensing Systems

In acoustic sensing system, in-the-ear or on the neck regions are primarily instrumented with sensors to detect sounds associated with chewing and swallowing, where microphones are widely used. Amft et al. [8] investigated the chewing sound collected from a microphone located inside

the ear canal to classify four food types. The system recognized eating activity from non-eating activities with at most 99% accuracy and achieved 80% up to 100% accuracy in classifying the food types for the isolated chewing events. However, the acoustic based approaches for detecting chewing suffers from environmental acoustic noise, therefore several studies investigated proximity sensors to measure the deformation in the ear canal walls while chewing. Sazonov and Fontana [101] used a piezoelectric strain gauge positioned below the ear to monitor jaw movements produced during chewing and a small microphone located over the laryngopharynx to capture chewing sound. Liu et al. [68] developed a food logging application to capture audio and first-person point-of-view images. The system processes all incoming sounds in real time through a head-mounted microphone and a classifier identifies when chewing is taking place, prompting a wearable camera to capture a video of the eating activity. The work was validated by the technical feasibility of the method with a small user study. On the other hand, Paßler and Fischer [88], Paßler et al. [89] investigated the use of additional reference microphones to eliminate environmental noise while monitoring chewing sound.

Swallowing involves contraction and relaxation of muscles of the tongue, pharynx and esophagus while food or liquid is passed from the mouth to the stomach [24]. Microphones can be placed in the ear or on the throat, and surface electromyography can be used to monitor muscle contractions and relaxations to capture swallowing. Amft and Troster [9] integrated gel electrodes and electret condenser microphone placed around the neck to recognize swallowing activity by surface EMG and sound signals. Olubanjo and Ghovanloo [84] automatically detected swallowing in real-time from the acoustic signals captured by a throat microphone placed over the suprasternal notch of the trachea, with an overall recall of 79.9% and precision of 67.6%. Several neck-worn systems have been studied for swallowing detection and recognize eating and non-eating activities by sounds produced in the user's throat area [84, 128]. Recently, deep learning is used to detect chewing and swallowing while eating [37, 57]. Gao et al. [37] proposed a Bluetooth headset to capture chewing sound and detect eating episodes using Deep Boltzmann Machine, with an accuracy of 94.72% for in-the-field testing. In general, the disadvantage of acoustic-based systems is its sensitivity to environmental and background noise. Therefore the majority of the aforementioned studies collected data in the lab or under controlled conditions. Table 1.1 summarizes several characteristics of the studies mentioned here.

Year	Publication	Sensors	Subjects	Objectives
2005	Amft et al. [8]	microphone	4	Distinguished between a small predefined set of different food types from chewing sound.
2008	Nishimura and Kuroda [82]	microphone	-	Proposed robust chewing number counting algorithm.
2010	Lopez-Meyer et al. [70]	microphone	18	Discriminated between swallow events.
2010	Amft [6]	earpad sensor	2	Classified 19 food type based on chew sounds.
2010	Shuzo et al. [109]	microphone	5	Count the number of chewing. Discriminated between 4 eating types (eating a hard food, eating a soft food, drinking water, speaking).
2011	Walker and Bhatia [124]	microphone	2	Discriminated between swallow, vocal chord activation, clearing of throat, coughing.
2012	Sazonov and Fontana [101]	piezoelectric strain gauge sensor	20	Detected periods of chewing/non-chewing events.
2012	Paßler et al. [89]	microphone	50	Classified 7 food types and 1 drink by adapting chewing sound models.
2012	Liu et al. [68]	microphone, camera	6	Detected chewing sound and combined with images captured from camera to provide dietary information.
2012	Yatani and Truong [128]	microphone	10	Classified 12 activities, such as eating, drinking, laughing and coughing by the sounds recored in the user's throat area.
2014	Olubanjo and Ghovanloo [84]	throat microphone	6	Detected acoustic-based real-time swallowing.
2014	Paßler and Fischer [88]	microphone	-	Captured chewing sound by a microphone located in the outer ear canal and detected chewing activity by eight algorithms in the presence of environmental sounds.
2016	Gao et al. [37]	Bluetooth headsets	28	Monitored chewing sound by the Bluetooth headsets to detect eating activity from non-eating activities using a deep learning algorithm.

Table 1.1: Summary of acoustic sensing systems.

1.4.2 Camera-based Sensing Systems

Camera-based sensing systems monitor eating activities by analyzing images or videos captured while eating. Unlike methods that investigated swallowing or chewing as the proxy for eating detection, camera-based sensing takes advantage of photos or videos during eating and utilizes computer vision algorithms to estimate the consumed food amount. Besides, the system has been successfully employed for recording the ground truth of eating activity [12, 15]. The accuracy of intake estimation is determined by two factors: object detection and volume estimation. Multiple classification algorithms such as Support Vector Machine (SVM) [115] are used to recognize food types, followed by food calibration and volume calculation to estimate EI [50, 122]. Martin et al. [75] proposed a method called Remote Food Photography Method (RFPM) to estimate food intake from images sent before and after each meal in free-living conditions. Puri et al. [95] analyzed food images taken at different positions captured by mobile phones to recognize 150 food types and estimate 3D food volume, with the quantitative nutrition information returned to the mobile phone. A card was used as the reference for the viewpoint and distance of the camera in images with image's color, scale, and orientation analyzed, and food was segmented and classified for volume estimation. Pettitt et al. [92] proposed a system to record video images of food consumed using an ear-worn micro-camera to estimate energy intake and dietary intake assessment. However, these methods ask the individuals to capture images before sending to the server or researchers for analysis, or control the on/off switch of the camera while eating, which is inconvenient and require the compliance of the individuals.

Recently, several studies have been investigating the camera-based dietary monitoring system that can automatically capture images while eating, such as SenseCam [45] and eButton [114], without human attention. SenseCam, originally developed by Microsoft, is a lightweight digital camera worn around the neck that automatically captures first-person point of view images and sensor readings at regular intervals throughout the day [45]. Gemming et al. [38] assessed the context of eating episodes captured by SenseCam. eButton, a miniature computer which has the similar functionality of SenseCam, can be worn like a chest button to passively capture images while eating, without interrupting the participant's eating behavior [114]. Jia et al. [50] conducted dietary assessment using eButton. In this study, images of 100 food samples were collected and each food volume was estimated to evaluate the accuracy of the calculated food portion size from eButton pictures,

Year	Publication	Sensors	Subjects/Images	Objectives
2008	Martin et al. [75]	camera	52 (subjects)	Estimated food intake based on picture sent before / after each meal.
2009	Puri et al. [95]	camera	13K (images)	Recognized food types, estimated consumed volumes and reported quantitative nutrition information of 6 groups of food.
2012	Almaghrabi et al. [5]	camera	100 (images)	Proposed a food recognition system which was coupled with nutrition tables to obtain energy intake estimation in a small data set. The system requires the user to point at the food to start the process.
2014	Sun et al. [114]	miniature computer	-	Proposed a chest-worn electronic device (eButton) for continuous monitoring of health, safety and wellbeing.
2014	Jia et al. [50]	miniature computer	7 (subjects)	Evaluated the accuracy of the calculated food portion size (volumes) from eButton pictures, compared by manual estimation of human raters.
2015	Meyers et al. [77]	camera	101K (images)	Recognized contents of food and estimated nutritional contents, such as calories.
2015	Gemming et al. [38]	camera	40 (subjects)	Analyzed images of participants' eating episodes captured by a digital camera and accessed the environmental and social context that surrounds eating and dietary behaviours.
2016	Pettitt et al. [92]	micro-camera	6 (subjects)	Investigated dietary intake assessment using a lightweight, wearable micro-camera.
2017	Doulah and Sazonov [28]	camera	7 (subjects)	Clustered food into food and non-food groups based on histogram matching from images captured by a wearable camera.
2017	Liang and Li [67]	camera	2978 (images)	Released a food image dataset and proposed an estimation method of consumed food calorie using a deep learning method (Fast R-CNN).

Table 1.2: Summary of camera-based systems.

with -2.8% of mean relative error between the estimated volume and the actual volume. Table 1.2 summarizes several characteristics of the studies mentioned here.

1.4.3 Motion-based Sensing Systems

In motion based sensing systems, inertial sensors are mounted on different locations of the body to detect eating activities. Several studies investigated the use of hand/wrist-worn wearable devices with accelerometers, gyroscopes and smart watches to detect gestures related to eating. Amft and Troster [10], Junker et al. [52] used five inertial sensors placed on the wrists, upper arms and on the upper torso to capture eating gestures. Zhang et al. [135] investigated a kinematic model of forearm movements to recognize eating and drinking gestures, with accelerometers located on the wrists, features extracted by an extended Kalman filter and classifier using hierarchical temporal memory network. However, only eating and drinking gestures were analyzed so it is unclear about the systems' ability to recognize other eating related activities. Kim et al. [56] investigated recognition of 29 predefined eating activities of Asian style w.r.t. spoon, chopsticks and hands, and food types using wrist-band accelerometers. The study obtained the recognition of an average F-measurement of 21% but failed to classify the hand actions. Thomaz et al. [118] investigated identifying eating moments using 3-axis accelerometer sensor data from an off-the-shelf smartwatch. 11 gestures of 20 subjects were recognized with eating and non-eating related activities and eating moments were estimated when a minimum number of inferred intake gestures were within a certain temporal distance of each other. The study obtained F scores of 76.1% and 71.3% for two free-living conditions (7 participants, 1 day; 1 participant, 31 days) and is promising for practical eating detection. However, a fixed size of gesture was used to compute features within the gesture and so it is unclear about the system's ability to recognize gestures with variable duration.

Recently, smart eyeglasses based sensing systems have been widely investigated. Google Glass has been used for automatic eating detection [96, 129]. A research group proposed a 3D-printed smart eyeglasses with EMG electrodes to monitor temporalis muscle's activity, detect chewing and eating events [131, 132, 133]. Food hardness was analyzed on chewing EMG and classified the selected 3 food types with an accuracy of 94.7%, with approximately 80% recall and precision achieved of chewing detection in fully unconstrained daily life. Farooq and Sazonov [31] monitored chewing cycles by attaching a piezoelectric sensor on the temporalis epidermis in the form of eyeglasses, which can monitor eating activity even in walking condition. Chung et al. [22] proposed a wearable

Year	Publication	Sensors	Subjects	Objectives
2008	Junker et al. [52]	microphone, surface EMG, elongation sensor	4	Segmented gesture motions and classify into 10 types with four gestures related to eating (cutlery, drink, spoon, handled).
2009	Zhang et al. [135]	accelerometers	-	Recognized eating and drinking gestures by tracking wrist motion.
2009	Dong et al. [25]	accelerometers, gyroscope	10	Detected in real-time information concerning bites taken during a meal.
2012	Kim et al. [56]	accelerometers	13	Recognized 29 pre-defined Asian style eating activities.
2012	Dong et al. [26]	accelerometers, gyroscope	49	Expanded #participants and food varieties in [25] to detect bite/drink events.
2015	Thomaz et al. [118]	accelerometers	7	Recognized 11 gestures and estimated eating moments.
2015	Rahman et al. [96]	Google Glass (accelerometers, gyroscope, magnetometer)	38	Classified eating activity from non-eating activities based on head movement data.
2016	Zhang et al. [133]	EMG electrode	8	Proposed a 3D printed smart eyeglass to detect chewing events and classificy food texture under controlled conditions.
2016	Farooq and Sazonov [31]	piezoelectric strain sensor, accelerometer	10	Captured signals collected by a combination of piezoelectric strain sensor, accelerometer and Bluetooth connected to the temple of glasses to detect periods of food intake in the presence of physical movements.
2017	Shen et al. [107]	accelerometers, gyroscope	271	Further expanded #participants and food varieties (374) in [26] to detect bite/drink events.
2017	Chung et al. [22]	load cells	10	Detected facial signals to detect eating activity from talking, head movement and wink.
2018	Zhang and Amft [132]	EMG electrode	10	Improved the design in [133] and evaluated a meal detection based on 1-minute chewing rate estimates under free-living condition.

Table 1.3: Summary of motion-based sensing systems.

system (GlasSense) to recognize facial activities by monitoring temporalis muscles. Two load cells were integrated in GlasSense in the form of a 3D printed eyeglasses to recognize chewing, talking, head movement and wink. This system obtained an average F1 score of 94% to classify six activities.

In previous work our group developed a method that detects a pattern of wrist motion during the ingestion of a bite [25, 26]. An experimental evaluation of 49 people eating a meal of their choice in a laboratory setting found that the method counted bites with a sensitivity (ratio of true detections to total actual bites) of 86% and a positive predictive value (ratio of true detections to true detections plus false positives) of 81% [26]. The experiment also revealed that an inexpensive MEMS gyroscope was as accurate as a more sophisticated magnetic, angular rate and gravity (MARG) sensor in tracking the relevant motion pattern [26]. These experiments were conducted using wrist-worn devices that were tethered to a stationary computer in order to facilitate the recording of raw motion data. Subsequently, the method was instantiated in a wearable version that resembles a watch. The watch executes the algorithm to detect the relevant motion pattern on a microcontroller. A button is pressed at the beginning of an eating activity (e.g. meal or snack) to begin bite counting, and pressed again at the end of the eating activity to end bite counting. The total bite count for the eating activity is stored for subsequent downloading to an external computer. To test its relevance for measuring energy intake, 77 people wore the device for 2 weeks and used it to automatically count bites during all eating activities [105]. Participants completed the automated self-administered 24 hour recall to measure kilocalories consumed [113]. A total of 2,975 eating activities were evaluated, an average of 39 per participant. A comparison of automated bite count to kilocalories found an average per-individual correlation of 0.53, with 64 participants having a correlation between 0.4 and 0.7 [105]. This range of correlation is similar to what has been found in evaluations of energy expenditure measured by accelerometer-based devices (pedometers, physical activity monitors). Table 1.3 summarizes several characteristics of the studies mentioned here.

1.4.4 Multimodal-based Sensing Systems

Compared to unimodal sensing systems, multimodal sensing systems take advantage of multiple sensors from different perspective of on-body locations and are expected to be more robust for dietary monitoring. Examples of sensor fusions that have been investigated include the combination of microphone for detecting chewing/swallowing sound and the accelerometers for detecting hand/wrist motions, or EMG sensors for monitoring temporalis muscle's activity while eating. Bedri

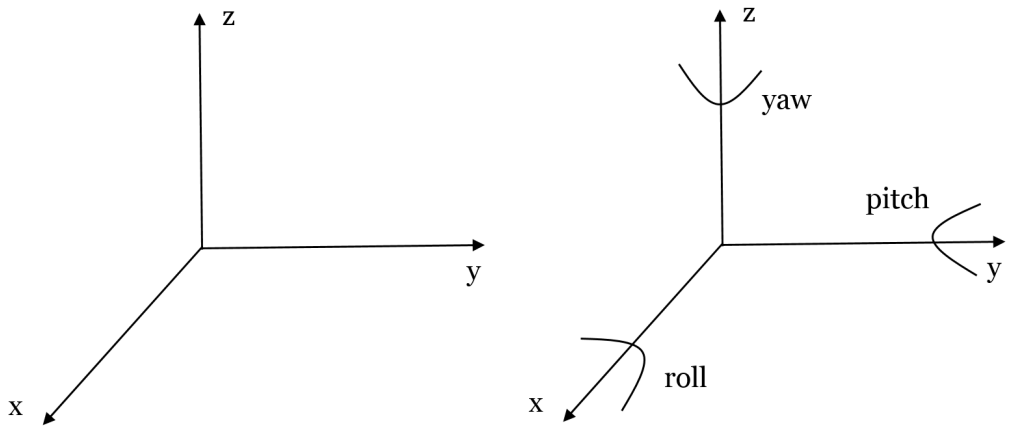
Year	Publication	Sensors	Subjects	Objectives
2015	[129]	Google Glass, Pebble Watch (accelerometers)	10	Combined signals collected from Google Glass and Pebble Watch to detect head motion from chewing and to detect hand-to-mouth (HtM) gestures.
2015	[108]	smartwatch, smartphone	5	Recognized 13 different activities including eaing, drinking, writing, jogging, etc. with a combination of smartphone and smartwatch.
2016	[76]	accelerometer, microphone	6	Fused audio sensor with motion sensors mounted on head and both wrists to recognize eating g in realistic scenarios.
2017	[16]	microphone, EMG sensors	20	Compared individual acoustic and EMG sensing and combined them to detect eating activity.
2017	[15]	microphone, in-ear proximity sensor, inertial motion unit	10	Recognized chewing activity at a 1-second resolution from signals collected from a novel sensing system (EarBit).
2018	[21]	accelerometer, proximity sensor	32	Captured jawbone movements to detect eating episodes in a controlled laboratory study, a controlled field study, and an in-the-wild study.

Table 1.4: Summary of multimodal-based sensing systems.

et al. [15] introduced a multi-modal wearable system (Earbit) including microphone around the neck, in-ear proximity sensor, 9 Degree-of-Freedom IMU (inertial motion unit) to detect eating events in relatively unconstrained environment. The system obtained an accuracy of 90.1% and an F1-score of 90.9% in the semi-controlled lab study to detect chewing instances, at a 1-second resolution and an accuracy of 93% and an F1-score of 80.1% in the unconstrained conditions. Study in [16] investigated the combination of acoustic and EMG signals to detect eating activity, with an accuracy of 91.5%, precision of 95.1% and recall of 87.4% in an uncontrolled-food condition. [21] designed an instrumented necklace combining accelerometer and range sensing to capture head and jawbone movements for detecting eating episodes. Three phases of experiments were conducted: a controlled laboratory study, a controlled field study, and an in-the-wild study, with the precision of 91.2%, 95.2% and 78.2% and the recall of 92.6%, 81.9% and 72.5%. Smart devices, such as smartphone, smartwatch and Google Glass are fused to monitor dietary behavior. Ye et al. [129] incorporated Google Glass and Pebble Watch to detect head motion from chewing and detect hand-to-mouth (HtM) gestures when eating. Shoaib et al. [108] fuse a smartwatch and a smartphone to recognize 13 different activities, such as eating, drinking coffee, walking upstairs, walking downstairs, sitting, etc, with the results showing that complex activities such as eating and drinking are recognized with a higher accuracy by combining sensors from the smartphone in the pocket position and a smartwatch. Merck et al. [76] combined head and wrist motion (Google Glass, smartwatches on each wrist), with audio signal (custom earbud microphone) to detect eating activity, with a precision of 92% and recall of 89% in detecting meals and detecting intakes by motion sensing signals. Table 1.4 summarizes several characteristics of the studies mentioned here.

1.5 Micro-electromechanical Systems Sensors (MEMS)

The tools described in this work make use of micro-electro-mechanical systems (MEMS) sensors, a class of devices that builds very small electrical and mechanical components on a single chip, to track wrist motion while eating. The advantages of MEMS sensors, including small size and low power consumption, make MEMS sensors widely used in monitoring of physical and eating activities. Typical sensor types include accelerometers, gyroscopes, microphones, electrocardiography sensors (EEG), magnetometers and optical sensors. This work makes use of accelerometers and gyroscopes.



(a) Accelerometer of X, Y, Z axis. (b) Gyroscope of yaw, pitch and roll axis.

Figure 1.4: MEMS accelerometer and gyroscope.

1.5.1 Accelerometers

An accelerometer is an electromechanical device used to measure acceleration forces. Such forces may be static, like the continuous force of gravity or, as is the case with many mobile devices, dynamic to sense movement or vibrations. Figure 1.4a displays the acceleration along x, y and z axis. Accelerometers are typically used in the following modes:

1. An inertial measurement of velocity and position;
2. A sensor of inclination, tilt, or orientation in 2 or 3 dimensions, as referenced from the acceleration of gravity ($1\ g \approx 9.81\ \mathrm{m}/s^2$);
3. A vibration or impact (shock) sensor.

When used in the field of wrist motion tracking, an accelerometer measures the linear motion of the wrist.

The basic principle of accelerometer is based on Newton's second law of motion and Hooke's law in Figure 1.5. According to Newton's second law of motion, a force on an object follows Equation 1.1:

$$F = m \times a \tag{1.1}$$

where F is the force, m is the mass of object and a is the acceleration.

According to Hooke's law, the displacement of a spring is proportional to the force applied

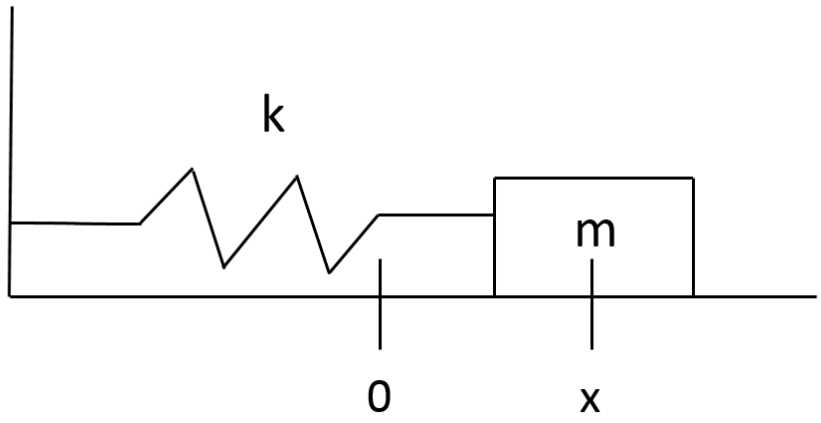


Figure 1.5: A mass spring system.

to it, as shown in Equation 1.2.

$$F = -k \times x \tag{1.2}$$

where F is the force, k is the spring constant, x is the displacement of the spring's current position w.r.t. its equilibrium position and negative acceleration refers to the compression of the spring. By combining these two equations, the acceleration can be obtained in Equation 1.3.

$$a = -\frac{k \times x}{m} \tag{1.3}$$

1.5.2 Gyroscopes

A gyroscope is a device that measures the angular velocity. Its design consists of a freely-rotating disk called a rotor, mounted onto a spinning axis in the center of a larger and more stable wheel. As the axis turns, the rotor remains stationary to indicate the central gravitational pull and hence the angular velocity can be measured. We use gyroscope to measure angular velocity along 3 axes of yaw, pitch and roll, which were initially used to describe an aircraft rotation. Yaw axis has its origin at the center of gravity and is directed towards the bottom of the aircraft. A yaw rotation is a movement around the yaw axis of a rigid body that changes the direction it is pointing, to the left or right of its direction of motion. Pitch axis has its origin at the center of gravity and is directed to the right. A pitch rotation is a movement up or down about an axis running from wing to wing. Roll axis has its origin at the center of gravity and is directed forward. A roll rotation is a

movement that lifts the left wing and lowers the right wing or vice versa. Figure 1.4b displays the rotational motion of yaw, pitch and roll.

The principle of gyroscope is based on Coriolis effect shown in Equation 1.4.

$$F = -2m\Omega \times v \tag{1.4}$$

where F is the force, m is the object mass, Ω is the angular velocity and v is the object velocity.

The main difference between the accelerometer and gyroscope is that gyroscope can measure rotation, whereas accelerometer cannot. In a way, the accelerometer can gauge the orientation of a stationary item with relation to Earth's surface. When accelerating in a particular direction, the accelerometer is unable to distinguish between that and the acceleration provided through Earth's gravitational force. The gyroscope maintains its level of effectiveness by being able to measure the rate of rotation around a particular axis. When gauging the rate of rotation around the roll axis of an aircraft, it identifies an actual value until the object stabilizes out. Using the key principles of angular momentum, the gyroscope helps indicate orientation. In comparison, the accelerometer measures linear acceleration based on vibration.

1.5.3 Bite Counter Device

In this work, a custom wrist-worn device based bite counter containing MEMS accelerometers (STMicroelectronics LIS344ALH) and gyroscopes (STMicro-electronics LPR410AL) was used to record the wrist motion of each participant at 15 Hz. Detailed description of the device can be found in [46]. A brief description of the device is provided here. Accelerometer used STMicroelectronics LIS344ALH to measure three-axis linear acceleration. The measurement unit is gravity units (g). Other than gravity, the accelerometer measures a deviation from free fall i.e. if the sensor is laying on a horizontal surface this will measure 0 g in the X and Y axis whereas the Z axis will measure 1 g. Gyroscopes used STMicro-electronics LPR410AL to measure three-axis rotational velocity, i.e. along yaw, pitch and roll axes. The measurement unit is degrees per second. A voltage signal is output by each gyroscope, with 2.5 mV representing 1 degree per second. Figure 1.4 displays the example of accelerometer and gyroscope.

1.6 Hidden Markov Models

This section is adapted from [2, 61] to introduce the principles of HMM. HMM is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (i.e. hidden) states. Markov process is a stochastic process that satisfies the Markov property. A stochastic process has the Markov property if the conditional probability distribution of future states of the process (conditional on both past and present states) depends only upon the present state, not on the sequence of events that preceded it, as illustrated in Equation 1.5.

$$P(X_n = x_n | X_{n-1} = x_{n-1}, ..., X_0 = x_0) = p(X_n = x_n | X_{n-1} = x_{n-1}) \tag{1.5}$$

where $X = (X_t : t \geq 0)$ is a stochastic process.

In a Markov chain, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters, while in the hidden Markov model, the state is not directly visible, but the output (observations) dependent on the state is visible. Each state has a probability distribution over the possible observations. Therefore, the sequence of observations generated by an HMM gives some information about the sequence of states.

In an HMM there exists an observation sequence $O = o_1, o_2, ..., o_T$ and a state sequence $Q = q_1, q_2, ..., q_T$. An observation o_t can be discrete or continuous and an element of the state sequence q_t represents any of the N possible states. According to the Markov property that observations are independent from each other, we can obtain:

$$\begin{aligned} p(q_1, ..., q_T | o_1, ..., o_T) &= \frac{P(o_1, ..., o_T | q_1, ..., q_T) p(q_1, ..., q_T)}{p(o_1, ..., o_T)} \\ &\propto \prod_{i=1}^{T} p(q_t | q_{t-1}) \prod_{i=1}^{T} p(o_t | q_t) \end{aligned} \tag{1.6}$$

where $p(q_t|q_{t-1})$ and $p(o_t|q_t)$ are the state transition probabilities and observable probabilities.

1.6.1 Elements of an HMM

Here several basic elements in a HMM are introduced:

1) N, the number of states in the model. We denote the individual states as $S = s_1, s_2, ..., s_N$

and the state at time t as q_t.

2) M, the number of distinct observation symbols per state. We denote the individual symbols as $V = v_1, v_2, ..., v_M$.

3) The state transition probability distribution $A = a_{ij}$ where

$$a_{ij} = P(q_{t+1} = s_i | q_t = s_i), \quad 1 \leq i, j \leq N \tag{1.7}$$

4) The obervation symbol probability distribution in state j, $B = b_i(k)$, where

$$b_i(k) = P(v_k \text{ at t } | q_t = s_j), \quad 1 \leq j \leq N, \quad 1 \leq k \leq M \tag{1.8}$$

A continuous representation of the observations has the advantage of better capturing the underlying statistical model. Then, $b_i(k)$ has the form of a probability density function. One common probability density function used for a HMM is the Gaussian density:

$$\begin{aligned} b_j(o_t) &= \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(o_t - \mu_j)^2}{2\sigma_j^2}) \\ &= \frac{1}{(2\pi)^{d/2}|\Sigma_j|^{1/2}} exp(-\frac{1}{2}(o_t - \mu_j)'\Sigma_j^{-1}(o_t - \mu_j)) \end{aligned} \tag{1.9}$$

If there are M number of Gaussians to model the density function, then c_m is a weighting value with $\sum_{m=1}^{M} = 1$ and $b_{jm}(o_t) = N(o_t; \mu_{jm}, \Sigma_{jm})$ and

$$b_j(o_t) = \sum_{m=1}^{M} c_m b_{jm}(o_t) \tag{1.10}$$

5) The initial state distribution $\Pi = \pi_i$ where

$$\pi_i = P(q_1 = s_i), \quad 1 \leq i, j \leq N \tag{1.11}$$

Given appropriate values of N, M, A, B and Π, the HMM can be used as a generator to give an observation sequence

$$O = O_1 O_2 \quad ... \quad O_T \tag{1.12}$$

as follows:

1) Choose an initial state $q_1 = s_i$ according to the initial state distribution π.

2) Set t =1.

3) Choose $O_i = v_k$ according to the symbol probability distribution in state s_i, i.e., $b_i(k)$.

4) Transit to a new state $q_{t+1} = s_j$, according to the state transition probability distribution for state s_i, i.e., a_{ij}.

5) Set t = t + 1; return to step 3) if t<T; otherwise terminate the procedure.

For convenience, we use the compact notation

$$\lambda = (A, B, \pi) \tag{1.13}$$

1.6.2 Types of HMM

Two common architectures are used in HMM: ergodic and left-right [61]. In ergodic HMM or fully connected HMM, every state of the model could be reached (in a single step) from every other state of the model. For an $N = 4$ state model, this type of model has the property that every a_{ij} coefficients is positive and the state transition matrix can be:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{1.14}$$

In left-right model, the underlying state sequence associated with the model has the property that as time increases the state index increases (or stays the same), i.e., the states proceed from left to right. The left-right typeof HMM has the desirable property that it can readily model signals whose properties change overtime e.g., speech or action. The fundamental property of all left-right HMMs is that the state transition coefficients have the property

$$a_{ij} = 0, \quad j < i \tag{1.15}$$

i.e., no transitions are allowed to states whose indices are lower than the current state. Furthermore,

the initial state probabilities have the property

$$\pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases} \tag{1.16}$$

since the state sequence must begin in state 1 (and end in state N). Often, with left-right models, additional constraints are placed on the state transition coefficients to make sure that large changes in state indices do not occur, with the form:

$$a_{ij} = 0, \quad j > i + \delta \tag{1.17}$$

For example, if the value of δ is 2, i.e., no jumps of more than 2 states are allowed, the state transition matrix can be:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix} \tag{1.18}$$

1.6.3 Three basic problems in HMM

In this part, three basic problems in HMM are introduced and discussed.

1.6.3.1 Evaluation on observation sequence

Given a model and a sequence of observations, the problem is to compute $p(O_1, O_2, ..., O_T|\lambda)$. The observation probability can be directly computed as:

$$p(O|\lambda) = \sum_{q_1,...q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1,q_2} b_{q_2}(O_2) ... a_{q_{T-1}q_T} b_{q_T}(Q_T) \tag{1.19}$$

However, the complexity is $O(2T \cdot N^T)$ since at every $t = 1, 2, ..., T$, there are N possible states which can be reached (i.e., there are N^T possible state sequences), and for each such state sequence about $2T$ calculations are required for each term in the sum of Equation 1.19. Clearly a more efficient procedure is required to solve this problem. Actually there is an algorithm called

forward-backward algorithm: Consider the forward variable $\alpha_t(i)$ defined as:

$$\alpha_t(i) = p(O_1, ..., O_t, q_t = s_i|\lambda) \tag{1.20}$$

which can be solved inductively as follows:

1) Initialization:

$$\alpha_1(i) = \pi_i b_i(O_1), \quad 1 \leq i, j \leq N \tag{1.21}$$

2) Induction:

$$\alpha_{t+1}(j) = [\sum_{i=1}^{N} \alpha_t(i) a_{ij}] b_j(O_{t+1}), \quad 1 \leq t \leq T-1, \quad 1 \leq j \leq N. \tag{1.22}$$

3) Termination:

$$p(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i). \tag{1.23}$$

With the forward-backward algorithm, the complexity is $O(N^2T)$.

1.6.3.2 State sequence decoding

Given the observation sequence $O = O_1O_2, ..., O_T$, and the model λ, the problem is to choose the optimal state sequence that could best "explain" the observations. One solution is to use Viterbi algorithm [61]. To find the best state sequence $Q = q_1, q_2, ...q_T$ for the given observation $O = O_1, O_2, ..., O_T$ and the quantity is defined:

$$\delta_t(i) = max_{q_1, ...q_{t-1}} p(q_1, ..., q_t = i, O_1O_2, ...O_t|\lambda) \tag{1.24}$$

where $\delta_t(i)$ is the best score along a single path at time t.

The best states can be decoded as follows:

1) Initialization:

$$\begin{aligned} \delta_1(i) &= \pi_i b_i(O_1), \quad 1 \leq i \leq N \\ \psi_1(i) &= 0. \end{aligned} \tag{1.25}$$

2) Recursion:

$$\begin{aligned} \delta_t(j) &= max_{1\leq i\leq N}[\delta_{t-1}(i)a_{ij}]b_j(O_t), \quad 2\leq t\leq T, \quad 1\leq j\leq N \\ \psi_t(j) &= argmax_{1\leq i\leq N}[\psi_{t-1}(i)a_{ij}], \quad 2\leq t\leq T, \quad 1\leq j\leq N \end{aligned} \tag{1.26}$$

3) Termination:

$$\begin{aligned} P^* &= max_{1\leq i\leq N}[\delta_T(i)] \\ q_T^* &= argmax_{1\leq i\leq N}[\delta_T(i)]. \end{aligned} \tag{1.27}$$

4) Path (state sequence) backtracking:

$$q_t^* = \psi_{t+1}(q_{t+1}^*), t = T-1, T-2, ..., 1. \tag{1.28}$$

1.6.3.3 Parameter estimation

The most difficult problem of HMM is to determine a method to adjust the model parameters (A, B, π) to maximize the probability of the observation sequence given the model. One popular method is Baum-Welch method (or equivalently the EM (expectation-modification) method. First, we introduce a variable:

$$\begin{aligned} \xi_t(i,j) &= p(q_t = s_i, q_{t+1} = s_i|O, \lambda) \\ &= \frac{\alpha_t(i,j)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{P(O|\lambda)} \\ &= \frac{\alpha_t(i,j)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^N\sum_{j=1}^N \alpha_t(i,j)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)} \end{aligned} \tag{1.29}$$

We defined $\gamma_t(i)$ as the probability of being in state s_j at time t, given the observation sequence and the model, giving

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i,j). \tag{1.30}$$

A method for reestimating the parameters of the HMM can be accomplished by using Equations 1.29 and 1.30. The parameters can be estimated as:

$$\begin{aligned} \bar{\pi_i} &= \gamma_1(i) \\ \bar{a_{ij}} &= \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \\ \bar{b_j}(k) &= \frac{\sum_{t=1}^{T} \xi_t(j)}{\sum_{t=1}^{T} \xi_t(j)}. \end{aligned} \tag{1.31}$$

In Appendix A we provide a tutorial of using a MATLAB toolbox for modeling HMMs. Please refer the implementation details in the Appendix.

Chapter 2

Assessment of the Accuracy of the Bite Counting Method across Demographic and Food Variables

The experiment described in this chapter evaluates a non-HMM algorithm designed to detect a single gesture type. Although the gesture is called "bite" it includes all instances of food or liquid intake. It provides a baseline accuracy which is used to compare against the performance of HMM-based algorithms on the same data set. It also assesses its accuracy across demographic (age, gender, ethnicity) and bite (utensil, container, hand used, food type) variables. The work in this chapter was published in the Journal of Biomedical and Health Informatics [107].

2.1 Introduction

In previous work our group developed a method that detects a pattern of wrist motion during the ingestion of a bite [25, 26]. An experimental evaluation of 49 people eating a meal of their choice in a laboratory setting found that the method counted bites with a sensitivity (ratio of true detections to total actual bites) of 86% and a positive predictive value (ratio of true detections to true detections plus false positives) of 81% [26]. The experiment also revealed that an inexpensive micro-electro-mechanical systems (MEMS) gyroscope was as accurate as a more sophisticated magnetic, angular

rate and gravity (MARG) sensor in tracking the relevant motion pattern [26]. These experiments were conducted using wrist-worn devices that were tethered to a stationary computer in order to facilitate the recording of raw motion data. Subsequently, the method was instantiated in a wearable version that resembles a watch. The watch executes the algorithm to detect the relevant motion pattern on a microcontroller. A button is pressed at the beginning of an eating activity (e.g. meal or snack) to begin bite counting, and pressed again at the end of the eating activity to end bite counting. The total bite count for the eating activity is stored for subsequent downloading to an external computer. To test its relevance for measuring energy intake, 77 people wore the device for 2 weeks and used it to automatically count bites during all eating activities [105]. Participants completed the automated self-administered 24 hour recall to measure kilocalories consumed [113]. A total of 2,975 eating activities were evaluated, an average of 39 per participant. A comparison of automated bite count to kilocalories found an average per-individual correlation of 0.53, with 64 participants having a correlation between 0.4 and 0.7 [105]. This range of correlation is similar to what has been found in evaluations of energy expenditure measured by accelerometer-based devices (pedometers, physical activity monitors).

This chapter describes an experiment conducted to further evaluate the accuracy of the automated bite counting method. The goal was to record a large number of people eating a wide variety of foods and beverages to evaluate its accuracy in terms of demographic variables (gender, age, ethnicity) and bite variables (food type, hand used, utensil, container). One approach to such an experiment is to script activities and ask each participant to complete the script. For example, a participant could be asked to consume 5 bites of 20 different types of food in a controlled order. This approach has been taken in some other studies of eating activities (e.g. [9, 88, 101]). Advantages to this approach include limiting the set of food types, simplifying the ground truth identification of events due to the use of a controlled script, and ensuring an equal quantity of each event type through repetition. However, this is unnatural in terms of food choices, eating pace, food order, and overall behavior during normal eating. Instead, we instrumented a cafeteria setting. Participants were allowed to select their own foods and eat naturally. This resulted in unequal distributions of bite variables which is offset by recording a large number of participants.

Figure 2.1: The table instrumented for data collection. Each participant wore a custom tethered device to track wrist motion.

2.2 Methods

2.2.1 Instrumentation

The experiment took place in the Harcombe Dining Hall at Clemson University. The cafeteria seats up to 800 people and serves a large variety of foods and beverages from 10-15 different serving lines. Figure 2.1 shows an illustration and picture of our instrumented table [49]. It is capable of recording data from up to four participants simultaneously and is similar to others in the cafeteria so that its appearance would not be distracting. Four digital video cameras in the ceiling (approximately 5 meters height) were used to record each participants mouth, torso, and tray during meal consumption. A custom wrist-worn device containing MEMS accelerometers (STMicroelectronics LIS344ALH) and gyroscopes (STMicro-electronics LPR410AL) was used to record the wrist motion of each participant at 15 Hz. Cameras and wrist motion trackers were wired to the same computers and used timestamps for synchronization. All the data were smoothed using a Gaussian-weighted window of width 1 s and standard deviation of $\frac{2}{3}$ s:

$$S_t = \sum_{i=-N}^{0} R_{t+i} \frac{exp\left(\frac{-t^2}{2\sigma^2}\right)}{\sum_{x=0}^{N} exp\left(\frac{-(x-N)^2}{2\sigma^2}\right)} \tag{2.1}$$

2.2.2 Participants

The Clemson University Institutional Review Board approved data collection and each subject provided informed consent. A total of 276 participants were recruited and each consumed a single meal [99]. Participants were free to choose any available foods and beverages. Upon sitting at the table to eat, an experimental assistant placed the wrist motion tracking device on the dominant hand of the participant and interviewed them to record the identities of foods selected. The participant was then free to eat naturally. If additional servings were desired, the participant was instructed to notify the experimental assistant to assist with removing the wrist motion tracker before moving through the cafeteria to obtain more food or beverage, returning to the table to begin a new segment of recording. Each such segment is referred to as a course. For 5 participants, either the video or wrist motion tracking data failed to record, and so are excluded from analysis. Total usable data includes 271 participants, 518 courses with a range of 1-4 and average of 1.8 courses per participant. Demographics of the participants are 131 male, 140 female; age 18-75; height 50-77 in (127-195 cm); weight 100-335 lb (45-152 kg); self-identified ethnicity 26 African American, 29 Asian or Pacific Islander, 190 Caucasian, 11 Hispanic, 15 Other.

2.2.3 Ground truth

The goal of the ground truthing process was to identify the time, food, hand, utensil and container for each bite. Because our data set is so large and was collected during natural (unscripted) eating, the total process took more than 1,000 man-hours of work. Figure 2.2 shows a custom program we built to facilitate the process. The left panel displays the video while the right panel shows the synchronized wrist motion tracking data. Keyboard controls allow for play, pause, rewind and fast forward. The horizontal scroll bar allows for jumping throughout the recording and additional keyboard controls allow for jumping to previously labeled bites. A human rater annotates a course by watching the video and pausing it at times when a bite is seen to be taken, using frame-by-frame rewinding and forwarding to identify the time when food or beverage is placed into the mouth. Figure 2.3 shows an example of a sequence of images surrounding a bite. Once the bite time is identified, the rater presses a key to spawn a pop-up window that allows the user to select from a list of foods recorded as having been eaten by the participant during the course, and a list of hand, utensil and container options. The process of ground truthing a single course took 20-60 minutes.

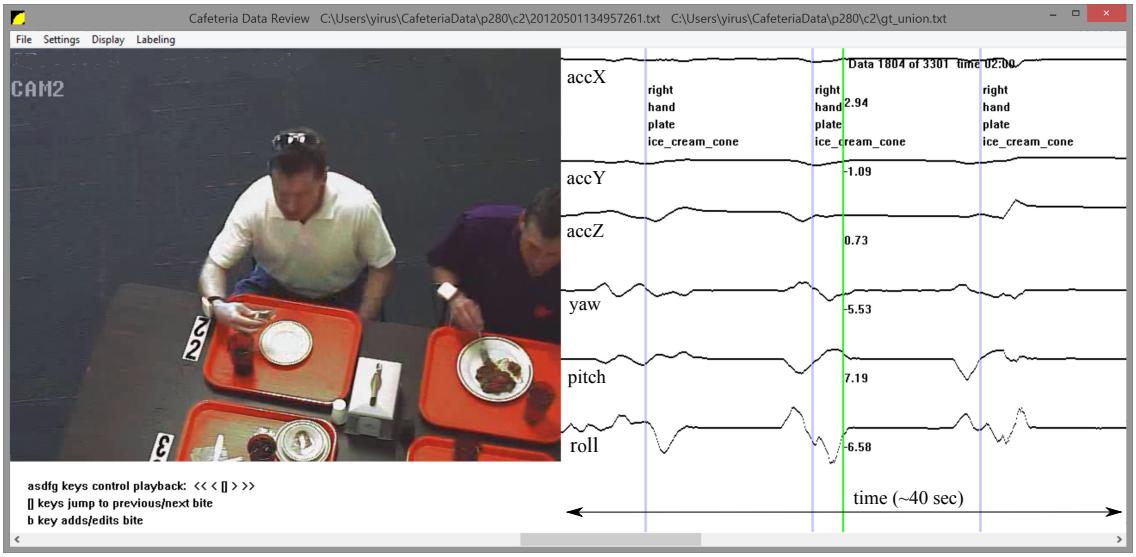


Figure 2.2: A custom program created for manual labeling of ground truth bites. The left panel shows the video and the right panel shows the wrist motion tracking. Vertical purple lines indicate the times marked as bites, the vertical green line indicates the time currently displayed in the video. Variables (hand, utensil, container, food) are identified for each bite.

(a) frame=0 (a) frame=7 (a) frame=14 (a) frame=21 (a) frame=28

Figure 2.3: Example identifying the time index of a bite (frame 14).

Figure 2.4: Examples of foods. From left to right: cheese pizza; cereal Apple Jacks; chunky chocolate chip cookie; California chicken wrap, shoestring french fries; hamburger, shoestring french fries.

In total, 374 different food and beverage types were chosen by participants. Food and beverage names were taken from the menus of the cafeteria. Some foods are given the generic name of the food line from which they are served due to the heterogeneous mixture of ingredients that could be custom selected by the participant, for example from a salad bar. In cases where a participant mixed 2 or more uniquely chosen foods, a single name was used that identified the combination. In cases where a participant ordered a custom version of a food in a food line, the modifier custom was included in the name. Example food identities include salad bar, shoestring french fries, Asian vegetables, pasta tour of Italy, cheese pizza, homestyle chicken sandwich, hamburger, custom sandwich, garlic breadsticks, fried shrimp and grapefruit. Example beverage identities include whole milk, coca cola, water, sweet tea, coffee and apple juice. Figure 2.4 shows some example images of foods. Foods and beverages were served in four types of containers: plate, bowl, glass and mug. Four different utensils were used: fork, spoon, chopsticks and hand. Hand could be identified as left, right or both.

Two human raters independently labeled each course. A total of 22 raters contributed. Raters were trained during a 1 hour training session to understand the process and how to use the program for labeling. Quantifying rater agreement is complicated because labeling is a two step process. First, each rater had to decide when bites occurred. Second, they had to quantify food, hand, utensil and container for each bite. Therefore we developed a two stage approach to determining rater agreement.

For each bite labeled by one rater, a ± 1 sec window was searched for a corresponding bite from the second rater. If the food identity, hand, utensil and container all matched, then the bite was considered matched and the time index was taken as the average of the time indicated by the two raters. If a corresponding bite was found within the window but one or more of the variables did not match, then the bite was reviewed by a third rater who judged which variable values were

Figure 2.5: Examples of foods that are difficult to identify bite by bite. From left to right: collard greens, macaroni and cheese, corn bread; edamame, jasmine rice, stir fry; char sui braised pork, brown rice, peas and carrots; pork chop suey with white rice, turkey sliced; Mexican rice, refried beans, roast pork loin.

(a) frame=0 (a) frame=15 (a) frame=30 (a) frame=45 (a) frame=60

Figure 2.6: Example of difficulty identifying the time index of a bite due to obscuring head motion.

correct. If no corresponding bite was found within the window, the third rater reviewed the bite to determine if it was missed by one of the raters or if it was off by more than 1 sec from a bite labeled by the other rater, in which case the third rater judged the correct time.

Using this process, rater performance can be evaluated using four metrics: mistaken identity (food identified incorrectly), time error (bite labeled more than 1 second from actual time), missed bite (the rater missed the bite completely) and data entry error (hand, utensil or container was mislabeled). Figure 2.5 shows some examples of foods that can be difficult to identify, for example when 2 or more foods of similar color and texture are served overlapping each other. Figure 2.6 illustrates an example of when the time of a bite can be difficult to determine due to the head of the participant obscuring the precise time of food intake. Data entry errors occurred most commonly when a rater mistakenly labeled a bowl as a plate or a mug as a glass, either of which would propagate to all the related bites in the course. Table 2.1 summarizes the errors found as judged by the third rater.

The usefulness of a fourth rater independently labeling each course and then comparing it to the union judged by the third rater was explored. After 71 courses were labeled, the process was stopped. In those 71 courses the following total errors were found: 17 missed bites, 0 timing

missed bites	900 (3.7%)
time error	1217 (5%)
identity error	714 (3%)
data entry error	1059 (4.4%)

Table 2.1: Manual labeling error rates.

errors, 18 identity errors and 8 data entry errors (0.2% of the total bites). Given the large amount of time needed to independently label the data and the tiny amount of new errors discovered, it was determined that the quality of ground truth provided by two human raters and then judged by a third rater was sufficient.

2.2.4 Bite counting algorithm

The bite counting algorithm described in [26] is briefly repeated here for background. The algorithm detects a pattern of wrist roll motion associated with a bite through the detection of four events. First, the wrist roll velocity must surpass a positive threshold. Second, a minimum amount of time must pass. Third, the velocity must surpass a negative threshold. Finally, a minimum time must pass between the negative wrist roll for one bite and the positive wrist roll for the beginning of a next bite. The minimum times help reduce false positives during other motions. The algorithm for detecting a bite based on this motion pattern can be implemented as follows:

```
Let EVENT = 0
Loop
  Let Vt = measured roll vel. at time t
  If Vt > T1 and EVENT = 0
    EVENT = 1
    Let s = t
  if Vt < T2 and t-s > T3 and EVENT = 1
    Bite detected
    Let s = t
    EVENT = 2
  if EVENT = 2 and t-s > T4
    EVENT = 0
```

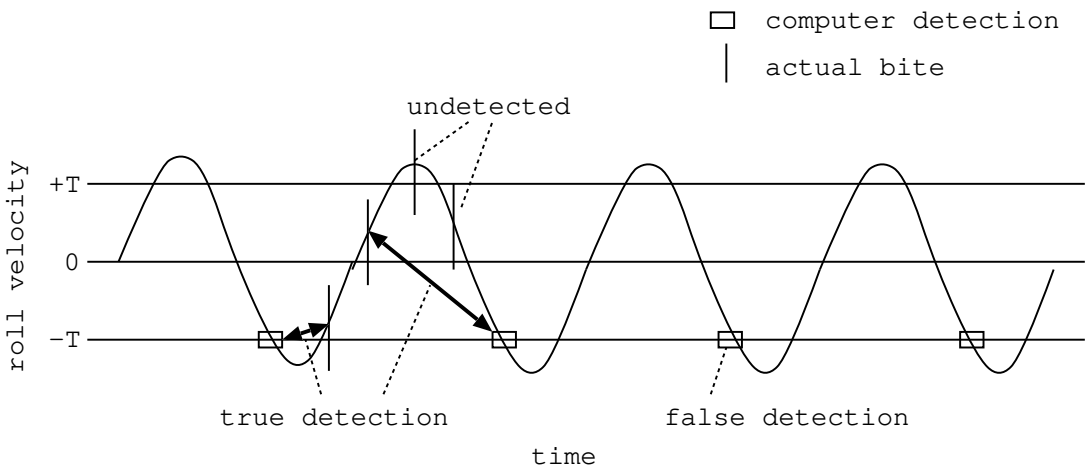


Figure 2.7: Classification of results.

The variable $EVENT$ iterates through the events just described and the parameters $T1$ and $T2$ define the threshold for roll detections.

2.2.5 Evaluation metrics

The evaluation method follows the procedure previously established [26]. Algorithm bite detections are compared to ground truth manually marked bites. Figure 2.7 illustrates the possible classifications. For each computer detected bite (small square in the figure), the interval of time from the previous detection to the following detection is considered. The first actual bite taken within this window, that has not yet been paired with a bite detection, is classified as a true detection (T). If there are no actual bite detections within that window, then the bite detection is classified as a false detection (F). After all bite detections have been classified, any additional actual bites that remain unpaired to bite detections are classified as undetected bites (U). This approach defines an objective range of time in which an actual bite must have occurred in order to classify a detected bite as a true positive. The window extends prior to the actual bite because it is possible in some cases for the wrist roll motion to complete just prior to the actual placing of food into the mouth. Accuracy (true detection rate) is calculated as (total Ts)/(total Ts+ total Us). Because this method does not allow for the definition of a true negative, specificity (false detection rate) cannot be calculated. We therefore calculate the positive predictive value as a measure of performance regarding false positives. The positive predictive value (PPV) is calculated as (total Ts)/(total Ts+ total Fs).

demographic	#partic.	#bites	#detected (%)	SPB
age				
51-75	21	1634	1404 (86%)	18
41-50	33	2790	2227 (80%)	17
31-40	27	2531	1949 (77%)	15
24-30	76	7426	5326 (72%)	13
18-23	114	9707	7050 (73%)	13
gender				
female	140	11811	9401 (80%)	15
male	131	12277	8555 (70%)	13
ethnicity				
African American	26	1958	1583 (81%)	18
Caucasian	190	15990	12327 (77%)	15
Hispanic	11	1195	877 (73%)	13
Other	15	1635	1115 (68%)	14
Asian or Pac. Isl.	29	3310	2054 (62%)	12

Table 2.2: Detection rate and seconds per bite (SPB) for age, gender, and ethnicity.

2.2.6 Parameter Tuning

In the original experiment involving 49 people eating a meal in a laboratory setting, $T1 = T2 = 10$, $T3 = 2$ and $T4 = 8$ were determined to be optimal [26]. It was also found that a range of values provided reasonable results. The present work reports results using these same values but also reports results using a shorter time for $T4$. During evaluation is was discovered that people ate faster on average in the cafeteria experiment than in the previous laboratory experiment. It was found that setting $T4 = 6$ produced a more balanced accuracy and positive predictive value. This is further discussed in sections 2.3-2.4.

2.3 Results

Table 2.2 lists the accuracies found across demographic variables age, gender and ethnicity. Accuracy trended higher as age increased. Females showed a 10% higher accuracy than males. The largest discrepancy observed was due to ethnicity, with African Americans showing the highest accuracy and Asians/Pacific Islanders showing the lowest accuracy. Table 2.2 also reports the average eating rate for each demographic in seconds per bite (SPB). SPB trends lower for every demographic as accuracy trends lower, suggesting that a faster eating rate results in lower accuracy.

Figure 2.8 plots the accuracy of the method for the foods of which more than 100 bites were

bite variable	#bites	#detected (%)
container		
bowl	3939	3091 (79%)
mug	116	87 (75%)
plate	16434	12389 (74%)
glass	3599	2389 (66%)
utensil		
fork	10308	8627 (83%)
spoon	2389	1711 (73%)
hand	10989	7419 (68%)
chopsticks	400	198 (50%)
hand used		
l-handed using left hand	1363	1106 (81%)
r-handed using right hand	18344	14267 (78%)
l-handed using both hands	162	116 (72%)
r-handed using both hands	1233	860 (70%)

Table 2.3: Detection rate and seconds per bite (SPB) for container, utensils, and hand used.

consumed. The average accuracy (75%) is given for reference. For most foods the accuracy trends consistently in the range of 60-90%. For a small number of foods the accuracy drops precipitously. For a food like ice cream cone the decrease in accuracy is likely due to the natural minimization of wrist roll during consumption (for fear of having the ice cream fall out of the cone). For other foods we manually observed the motion in the hundreds of hours of video to try to infer commonalities. In many cases a bite involves head-towards-plate motion in combination with hand-towards-mouth motion. The former seems to be larger when a food is more prone to spillage, so a participant positions their head over the container to facilitate delivery of the food to the mouth (for example, compare figure 2.3 to figure 2.6). To explore this hypothesis we calculated the amount of motion of the wrist during a 2 second window centered on every bite and took the average value for each food type, finding a modest correlation of 0.4.

Table 2.3 summarizes the accuracies found across other bite type variables. Container accuracy was fairly consistent with the exception of glass which was 9% lower than average. For utensils, chopsticks showed a relatively low detection rate (50%) but were also found to be used twice as fast (7 seconds per bite) as a fork or hand (14-15 seconds per bite). Handedness showed a small variation in accuracy, while the use of both hands as opposed to a single hand reduced accuracy by 8-9%.

Overall, across all 24,088 bites the accuracy was 75% with a positive predictive value of 89%.

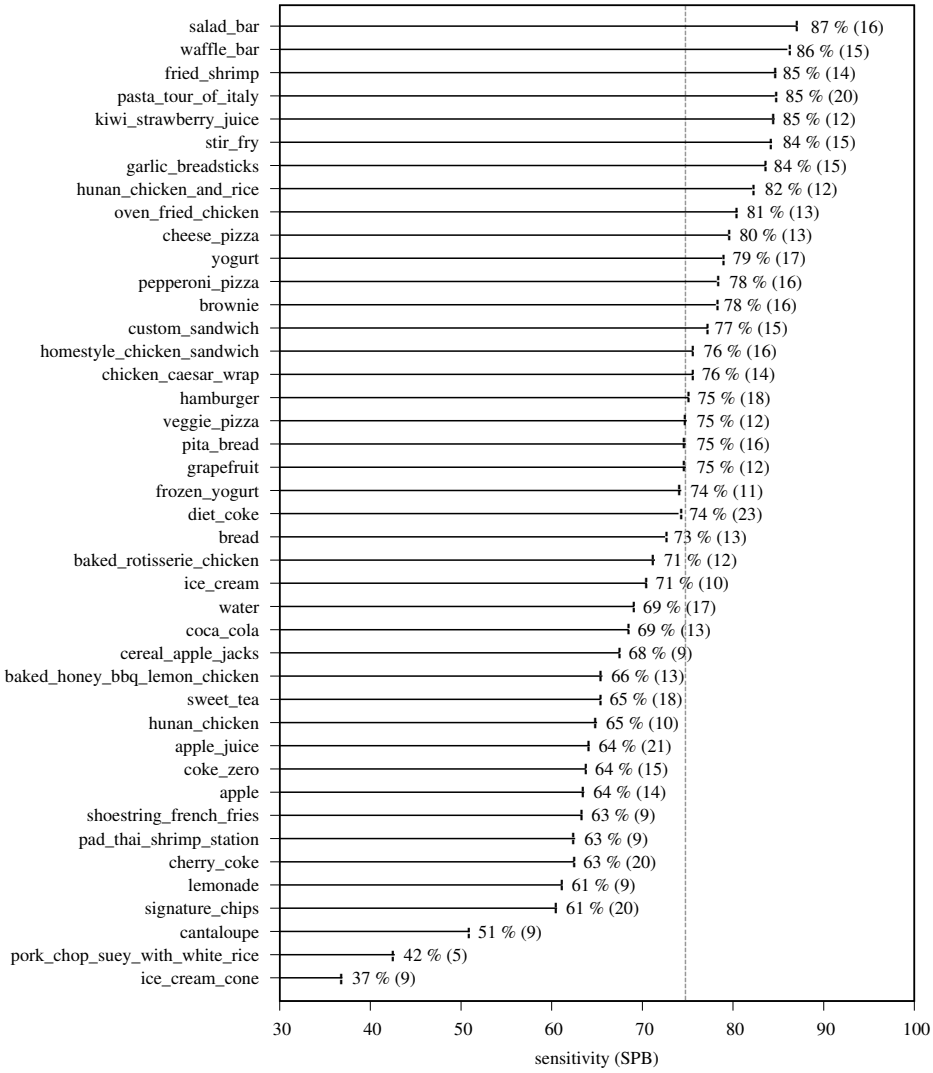


Figure 2.8: Detection rate for all foods of which participants consumed greater than 100 bites. Average detection rate (75%) highlighted for reference.

The algorithm parameters were originally determined using data recorded in a laboratory setting [26] in which the average eating rate was slower (n=49, seconds per bite = 19.1 ± 6.4) compared to what was observed in the cafeteria setting (n=271, seconds per bite = 14.7 ± 5.6). We therefore experimented with shortening the parameter controlling the minimum time between detections of bites to 6 seconds. With this value the algorithm produced 81% accuracy with a positive predictive value of 83%.

2.4 Conclusion

The primary goal of this chapter was to establish a baseline method to detect eating activities and assess its accuracy across a wide variety of demographics and food types. While minor variations occurred across most variables, the method showed robustness to this challenging data set. The original laboratory test found 81% accuracy with 86% positive predictive value [26]. After tuning the algorithm to the faster eating pace observed in the cafeteria, the same accuracy was achieved with only a 3% decrease in positive predictive value. This experiment provides the most comprehensive evidence to date that the method is reliable during normal unscripted eating.

The experiment identified two areas where the algorithm could be improved. First, variations in eating pace affect the accuracy. One parameter of the algorithm is designed to minimize false positives by requiring a minimum amount of time between detected bites. It may be possible to adjust this parameter in real-time while the algorithm is running similar to how a pedometer learns the stride duration of a person while running or walking and adjusts its step detection parameters accordingly. Second, variations in the amount of wrist motion versus the amount of head-towards-plate motion affect the accuracy. Two parameters of the algorithm are designed to detect the typical amount of motion. Again it may be possible to adjust these parameters in real-time to learn the typical amount of wrist motion of a person during a meal. This work provides the data set necessary to explore these ideas.

Studies have shown that participants change their eating behavior in clinical settings [23, 93]. As this method is intended to be used in free-living scenarios, a naturalistic evaluation of its accuracy is important. However, although we tried to make the cafeteria setting as natural as possible, it is still possible that behaviors in free-living environments could affect the accuracy of the method in ways that could not be captured with this study (e.g. grazing, other types of distraction).

Chapter 3

Lexicography of Hand Gestures During Eating

This chapter considers the problem of the lexicography of defining gestures a person makes while eating. We propose and test a vocabulary of actions to quantify gestural behaviors while eating based on discernible intent. The set of gestures include taking a bite of food (bite), sipping a drink of liquid (drink), manipulating food for preparation of intake (utensiling), and not moving (rest). All other activities such as using a napkin or gesturing while talking are grouped into a non-eating category (other). We test the lexicography by labeling segments of wrist motion according to the gesture set. This chapter describes detailed definitions of the gestures to inform human raters manually labeling the data.

3.1 Lexicography of Eating Gestures

A common lexicography of eating gestures is needed to support research in automated dietary monitoring. It helps research groups compare results and share data, measure progress, and identify areas where current methods fail. However, the lexicology in this domain is challenging due to the lack of standard definitions of terms defining actions one might take while eating. For example, a bite may refer to the action of placing food into the mouth for consumption, but may also refer to the compressive motion of the jaw on food already in the mouth. A drink may refer to

a single instance of beverage intake, or a full container of beverage. Several terms may be used to describe the manipulation of food prior to intake, such as cutting (to disassemble large pieces into smaller pieces), stirring (to combine foods), and dipping (to apply condiments). Some actions do not have standard terms, such as moving food onto a fork in preparation for bringing it to the mouth. In order to quantify eating behaviors it is necessary to establish objective, repeatable definitions. These can then be used to label ground truth for research into automatic segmentation and classification of wrist motion to quantify eating behaviors [97, 106].

Several methods have been used for the lexicography of gestures: top-down, bottom-up and the protocol used in the field of eating activity recognition.

3.1.1 Top-down Approach

In the top-down approach, meanings are defined first, with gestures designed to communicate the meanings. The most common example is sign language. Sign languages are a languages that use manual communication to convey meaning. This can include simultaneously employing hand gestures, movement, orientation of the fingers, arms or body, and facial expressions to convey a speaker's ideas. A large lexicon of both single-handed and two-handed gestures has been defined for common words and finger spelling for communication of obscure words or proper nouns [111, 112]. Another example is vision-based interfaces in video games, where sets of gestures have been defined to characterize commands of playing a game [54, 123]. One study [54] defined 10 intuitive gestures that were used as commands of a one-person action game, with each gesture reflecting an intuitive movement in the real world. A third example is traffic navigation, in which different hand and body gestures have been defined [110, 126]. Other examples can be found in human-computer interaction (HCI), in which gestures have been defined as the commands to help user interact with the computer [90, 98]. One study [90] lists a taxonomy of hand gestures for HCI, where meaningful gestures are differentiated from unintentional movements, and gestures used for manipulation of objects are separated from the gestures which posses inherent communicational character.

3.1.2 Bottom-up Approach

In the bottom-up approach, the problem is to define gestures describing common activities that are not intended to communicate meaning [18, 32, 62]. Motion sequences include walking,

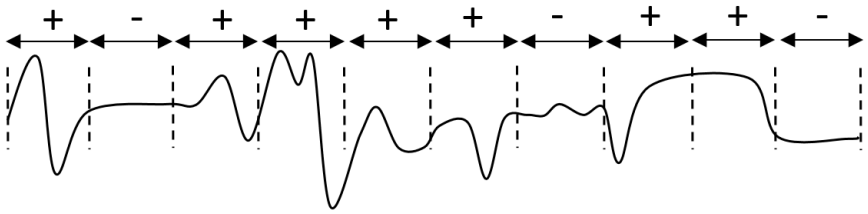
running, climbing in the field of ambulation, riding a bus, driving in the field of transportation, eating, drinking, reading, brushing teeth in the field of daily activities, rowing, spinning, lifting weights in the field of fitness, kneeling, situation assessment and opening a door in the field of military [62]. In the domain of daily activity recognition, a common lexicon includes walking, jogging, up/down stairs, cycling and similar physical activities [20, 60, 73, 78, 87, 127]. Independent of domains, [64] suggests some common rules for encoding hand gestures. Three modules were proposed in [64], which were kinetic gesture coding in which trajectory and dynamics of a hand movements were defined to represents a type of analysis that could be submitted to a 3-D video analysis, and bimanual relation coding in which the spatial relation and functional relation were defined to describe relation between left and right hand, and functional gesture coding in which gesture units that have been defined in kinetic gesture coding and bimanual relation coding were coded with respect to their function and were further classified as types such as hand-showing, deictic and body-deictic. This paradigm is followed in this work and described more later.

3.1.3 Approaches to Labeling Eating Activities

In the domain of eating activity recognition, three approaches have been taken to label data: window-based, index-based and segment-based. Figure 3.1 illustrates an example of each.

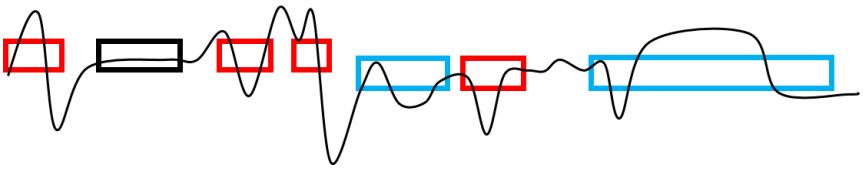
3.1.3.1 Window-based

In the window-based approach, equal intervals of data are labeled to indicate whether a particular type of eating event (for example, swallowing) occurs or not. Work in [100] labeled the events of chewing, swallowing and bite with 10 sessions each containing 3 parts of 20 min inactivity period, the meal period and a second 20 min inactivity period. Work in [85] recorded the eating and physical activities and manually partitioned each sensor file into consecutive intervals marked as eating or not. Inter-rater reliability was evaluated based on the number of counts of swallowing, chewing and bite in a fixed duration of data. An advantage of the window-based approach is that it simplifies the manual labeling of data. However, it does not indicate the exact time instant or duration of the event.



(a) Window-based. +/-: event occurs or not.

(b) Index-based. Vertical bar indicates event.



(c) Segment-based. Variable length of segments with different colors indicate different event types and durations.

Figure 3.1: Different approaches to annotating activity data during eating.

3.1.3.2 Index-based

In the index-based approach, a single time index is labeled when an event (for example, a bite or drink) occurs [26, 27, 88, 107]. Work in [88] recorded sound data of eating and annotated onset and end of every chew event of the intake cycle, with reference mark for a chew event at the center of the chew event label. Previous work in our group also labeled the index of eating and drinking activities in a cafeteria setting, along with utensil type, hand used, food type, container of the food [26, 27, 107]. This approach can be used to more precisely identify specific times of key events, but cannot describe events that occur across a range of time, such as cutting or stirring.

3.1.3.3 Segment-based

In the segment-based approach, sequences of time of variable duration are labeled, including the start and end index and the event type [7, 97, 106, 134]. Work in [134] labeled the start and end index of chewing with food (biscuit), water, cough and speak collected by smart EMG eyeglasses. Previous work in our group also investigated segment-based labeling for eating gestures, with start and end indices labeled for each gesture with variable durations [97, 106]. Segment-based approach provides more information but can be difficult to use for manual labeling as additional criteria must be determined by the human rater. This work uses segment based labeling for intake data.

3.2 Methods

3.2.1 Data

The same dataset in Chapter 2 was used here for the study of lexicography. A total of 276 participants were recruited and each consumed a single meal. For 5 participants, either the video or wrist motion tracking failed to record, and for 2 participants non-dominant hands were used for recording; these are excluded from analysis.

3.2.2 Definitions of Gestures

Our proposed lexicon was motivated by separating intake related gestures from non-intake related gestures. There are arguably fewer of the former compared to the latter. Since the primary

goal in research into automated monitoring of dietary intake is to quantify intake, our proposed lexicon uses a relatively small set of non-intake gestures.

We define four eating-related gestures (two intake and two non-intake): *bite*, *utensiling*, *drink*, *rest*. All other activities (e.g. gesturing while talking, cleaning with a napkin etc.) are referred to as a fifth gesture *other*. Following the paradigm proposed in [64], each gesture is defined consisting of the following parts with at least 1 second duration:

(a) the description of the activity;

(b) the start time of the activity;

(c) the end time of the activity;

(d) particular events that should be included or excluded;

Bite

(a) The subject puts food into their mouth.

(b) Starts when a hand or utensil starts moving towards the mouth.

(c) Ends when the hand or utensil finishes moving away from the mouth.

(d) Bites need not begin and end at a plate. Motion towards and away from the mouth should define the boundaries; with food consumption taking place in between.

(e) A single bite may include multiple successive back-and-forth motions from a utensil or hand to the mouth, that individually did not complete the hand motion away from the mouth, and that were separated by less than 1 second of time.

Drink

(a) The subject puts beverage into their mouth.

(b) Starts when a hand begins moving a beverage towards the mouth.

(c) Ends when the hand has finished moving away from the mouth.

(d) Each individual sip should be a different drink (if multiple sips are taken).

Utensiling

(a) The subjects uses an utensil or their hand(s) to manipulate, stir, mix or prepare food(s) for consumption.

(b) Starts when manipulating the food.

(c) Ends when manipulating has finished.

(d) This includes moving food around the plate, dipping foods in sauces, cutting foods, and other similar activities.

Rest

(a) The subject's dominant hand has little or no motion. The range of motion that may be considered rest depends upon the individual. Different people have different levels of physiological tremor (motion that occurs in everyone and has no medical significance) and thus the threshold for maximum motion during rest will vary subject to subject.

(b) The determination of rest should be based on the instrumented dominant hand only.

(c) Starts when there is no intent (subject's hand stop moving).

(d) Ends when new intent becomes apparent (subject's hand begins moving again with clear intent for at least 1 second).

(e) A period of rest may include time when a person is holding a utensil, food or drink, but where the instrumented hand is relatively motionless.

Other

(a) All other actions should be left unlabeled. Examples include reaching towards food (e.g. prior to a bite gesture), gesturing while talking, cleaning with a napkin, and moving a plate.

(b) In cases where the action of the instrumented hand alone is unclear, the subject's face and body can be viewed to help discern intent. For example, if the subject is talking and there is a slight motion in the instrumented hand, one may assume it is gesturing while talking (other) instead of rest.

(c) In cases where it is difficult to differentiate between rest and other, or utensiling and other, the other label is preferred.

3.2.3 Custom tool for gesture labeling

Figure 3.2 shows a custom program we built to facilitate labeling. The tool was coded using Microsoft Visual Studio. The left panel displays the video while the right panel shows the synchronized wrist motion tracking data. Top to bottom on the right panel shows the 6 axes of motion (AccX, AccY, AccZ, yaw, pitch and roll) with a seventh line at the bottom indicating tray weight as measured from a table embedded scale. Keyboard controls allow for play, pause, rewind and fast forward. Vertical green line indicates the time currently displayed in the video. A human rater annotates a meal by watching the video and uses frame-by-frame rewinding and forwarding to identify the start and end time and type of a gesture according to our definitions. Boxes laid over the seventh line indicate periods of time labeled as gestures (for example, red = bite). Unlabeled segments with duration longer than 4 seconds are considered as type other, unlabeled segments shorter than 4 seconds are considered transitions between gestures and are ignored [97]. The process of labeling a single meal took 60-120 minutes.

Due to the work-intensive nature of this labeling process, only 95 meals (20%) were labeled by two raters. In total 18 raters contributed to the process. Raters were trained in several training sessions to understand the process and the definitions of gestures.

3.2.4 Inter-rater Reliability

In order to test inter-rater reliability we developed methods to compare multiple labelings of the same meal recording. The data provided by raters is also intended to be used in the future for training classifiers and evaluating automatic segmentation, and thus the process for combining multiple raters' labels into a union is described. Since our data set is so large and was collected during natural (unscripted) eating, the total process took more than 700 man-hours of work.

Quantifying rater agreement is complicated because labeling contains multiple steps. First, each rater had to decide the start and end index of a gesture. Second, they had to identity the type of a gesture. Therefore we developed a custom approach that includes the following 6 cases to determining gesture matching as illustrated in Figure 3.3. For each gesture labeled by one rater, any overlapped gesture labeled from the second rater was examined.

Agreement.

If only one corresponding gesture with the same identity was matched and the disagreement of both

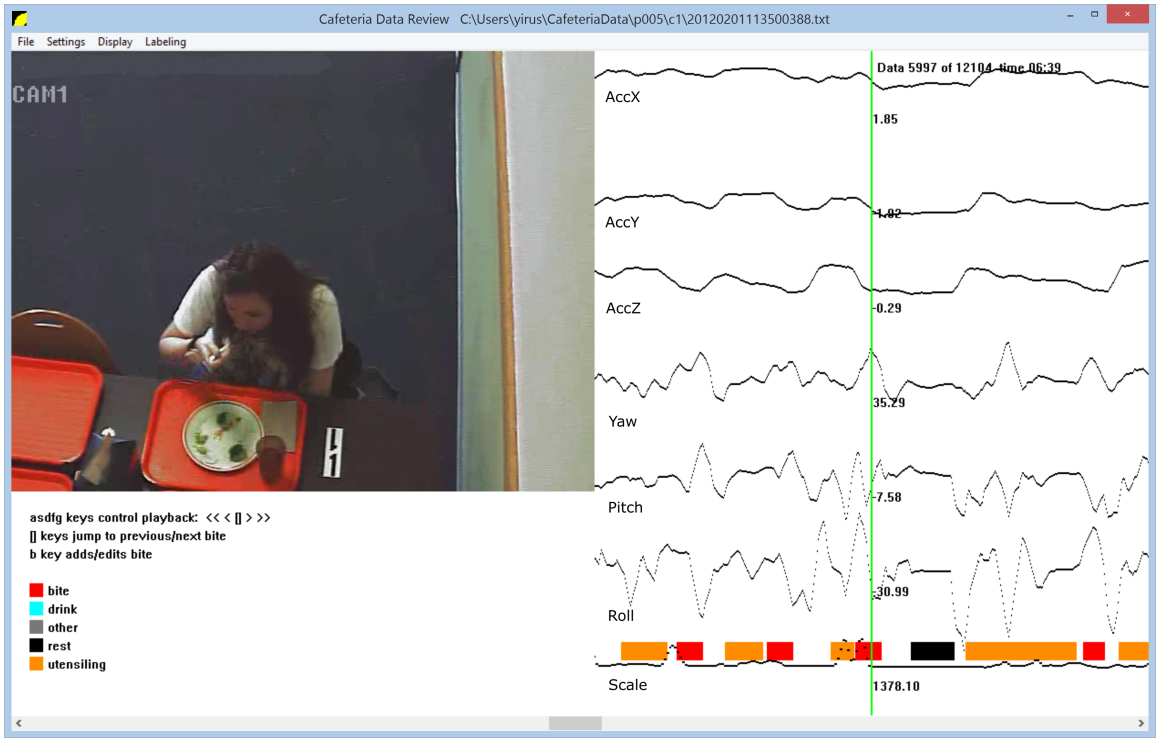
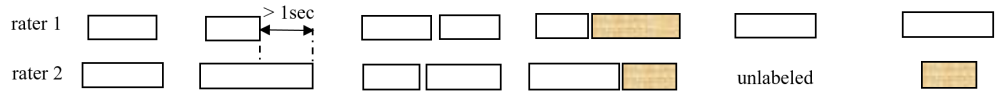


Figure 3.2: A custom program for gesture labeling. Box with different colors indicate gesture types: red = bite, aqua = drink, orange = utensiling, black = rest and grey = other.



(a) Agreement. (b) BA I. (c) BA II. (d) BA III. (e) Mistake-missed. (f) Mistake-identity.

Figure 3.3: Different cases of gesture matching between two raters. Segments with different colors represent different identities. BA: boundary ambiguity.

start and end index is within 1 second, then the start and end time index were taken as the averaged time indicated by two raters.

Boundary Ambiguity I.

If only one corresponding gesture with the same identity was matched and the disagreement of start or/and end index is longer than 1 second, and there are no other gestures labeled within the boundaries, then the start and end index were taken as in Equation 3.1.

$$t = \begin{cases} (t_1 + t_2)/2, & ||t_1 - t_2|| \leq 1 \text{ sec} \\ max(t_1, t_2), & \text{otherwise} \end{cases} \tag{3.1}$$

where t_1 and t_2 represent index labeled by rater #1 and rater #2, and max indicates the index providing the maximum gesture extent. For intake gestures, this is usually caused by a pause at the start or end of a gesture, e.g. during taking a bite the participant did not complete moving food towards the mouth until a pause for masticating food from a previous bite. For non-intake gestures, this is usually caused by some unintentional ambiguity in the definitions. For example, when dipping food in a sauce, one rater may label the motion of reaching towards the sauce as utensiling while a second rater starts the utensiling when the food touches the sauce.

Boundary Ambiguity II.

If multiple corresponding gestures with the same identity were matched, two cases are discussed here. For intake gestures, a corresponding match with single index-based labels created in chapter 2 were compared. Gestures from the rater which matched the index-based labels in terms of the amount of intake events were considered correct, and the longer duration of the two raters were taken as the union. For non-intake gestures, the max extent of the start and end index of the gesture was taken. Boundary Ambiguity II (N:N or N:1 matching) is usually caused by one rater labeling a single whole period of time as a single gesture while another rater segmented it into multiple sub-periods.

Boundary Ambiguity III.

If only one corresponding gesture with the same identity was matched and the disagreement of start or/and end index is longer than 1 second, and there is another gesture labeled within the boundaries but it was matched against a different gesture, then the start and end index were taken as in Equation 3.1. If the extra gesture did not match anything, then the gesture in query is considered as matched but the extra gesture is considered as mistake-identity.

Figure 3.4: Example of gesture matching. From top to bottom: gestures labeled by rater #1, rater #2 and the union. (a)-(f) illustrate different cases of gesture matching. Red = bite, aqua = drink, orange = utensiling, black = rest.

Mistake-missed.

If no corresponding gesture was matched, then the gesture in query was taken as the union. For intake gestures, this is usually caused by one rater missing an action. For non-intake gestures, this is usually caused by the ambiguity of definitions. For example, one rater may label rest for a participant while another rater may consider the same period of time as a gap.

Mistake-identity.

This happens when one corresponding gesture with a different identity was found. This is usually caused by a rater incorrectly identifying a gesture.

Using this process, rater performance can be evaluated using three metrics: agreement, boundary ambiguity (I, II, and III), and mistake (mistake-missed and mistake-identity). Figure 3.4 shows an example of gesture matching and the union labels.

3.2.5 Comparing Intake Gestures with Index-based Labels

In Chapter 2 we labeled this same data set with single time indices indicating when bites and drinks occurred. The time indices indicated when the food or beverage first touched the mouth initiating intake. We used this set of time indices to further test the segment labels. Note that this could only be done for intake gestures as events corresponding to utensiling, rest and other were not labeled in Chapter 2.

Each intake gesture (bite and drink) was searched for any corresponding index-based labels. If one gesture contained exactly one index-based label within its boundaries, then it was considered as agreement. If one gesture contained multiple index-based labels within its boundaries, then it was considered as ambiguity. This usually happened when a rater labeled one long segment that contained multiple short bites or drinks. If no index-based label was contained within a gesture, or

Type	#Gestures	Duration (sec)		
		Average ± Stddev	Min	Max
Bite	18462	2±1	1	11
Drink	2182	6±2	1	18
Utensiling	14861	5±5	1	186
Rest	14761	8±12	1	341
Other	1348	9±6	4	73

Table 3.1: Statistics of gestures.

an index-based label was not matched against any gesture, then it was considered as missed. Note that the index-based labels searched for matching intake gestures were only considered if they were performed by the dominant hands of the participants.

3.3 Results

Table 3.1 lists the distribution statistics of the five gestures labeled by all raters. The minimum durations of bite, drink, rest and utensiling were enforced to be 1 second; the minimum duration of other was enforced to be 4 seconds by definition. Notice that utensiling and rest could last up to 3 and 6 minutes, respectively. For utensiling, this happened when a subject took a long time peeling a tangerine. For rest, this happened when a subject talked to other people for a long time while the hand wearing device was at rest. The variable duration of gestures demonstrates the challenge of labeling segments: it is difficult to accurately label the start and end index.

Table 3.2 lists the inter-rater reliability for meals labeled by two raters. Note that only four gestures (bite, drink, rest and utensiling) are evaluated since gesture "other" will be automatically determined if the gap between gestures are longer than 4 seconds. The overall agreement is 92.5% with exact agreement of 75% and 17.5% of boundary ambiguity. The agreement for bite and drink is 99.4% and 98.1%, respectively. This indicates a high degree of agreement between raters on intake related gestures. The overall mistake rate is 7.5% with most of the mistakes from non-intake gestures. This is due to the nature of ambiguity on non-intake gestures. For example, raters may have different understanding on levels of physiological tremor in the definition of rest and one rater labeled a small amount of motion as rest while another rater did not label it.

The usefulness of a third rater independently labeling each meal and then comparing it to the union from two raters was explored. After 7 meals were labeled, the process was stopped. In those 7 meals, the mistake and ambiguity rate did not change.

Cases	#Gestures (%)	Bite (%)	Drink (%)	Rest (%)	Utensiling (%)
Agreement	13184 (75.0%)	5814 (89.6%)	594 (66.9%)	2926 (58.9%)	3850 (73.5%)
BA I	1030 (5.9%)	136 (2.1%)	160 (18.0%)	548 (11.1%)	186 (3.6%)
BA II	784 (4.5%)	21 (0.3%)	73 (8.2%)	443 (8.9%)	247 (4.7%)
BA III	1250 (7.1%)	478 (7.4%)	44 (5.0%)	232 (4.7%)	496 (9.5%)
Mistake -missed	1079 (6.1%)	23 (0.4%)	9 (1.0%)	700 (14.1%)	347 (6.6%)
Mistake -identity	249 (1.4%)	14 (0.2%)	8 (0.9%)	117 (2.4%)	110 (2.1%)
Overall mistake	1328 (7.5%)	37 (0.6%)	17 (1.9%)	817 (16.5%)	457 (8.7%)
Overall BA	3064 (17.5%)	635 (9.8%)	277 (31.2%)	1223 (24.6%)	929 (17.7%)
Overall agreement	16248 (92.4%)	6449 (99.4%)	871 (98.1%)	4149 (83.5%)	4779 (91.3%)
#Gestures	17576	6486	888	4966	5236

Table 3.2: Inter-rater reliability for meals with two raters. BA: boundary ambiguity.

	One rater	Two raters
Agreement (%)	16029 (94.4%)	3461 (95.8%)
Ambiguity (%)	93 (0.5%)	21 (0.6%)
Missed (%)	862 (5%)	128 (3.6%)
# Gestures	16984	3610

Table 3.3: Inter-rater reliability between intake gestures and index-based labels.

Table 3.3 lists the inter-rater reliability of comparing intake gestures with index-based labels. It can be seen that the agreement and missed rate were both improved by 1.4% when the second rater contributed to labeling. The agreement and mistake rate when a third rater contributed did not change compared to gestures labeled by two raters. The small amount of improvement of multiple raters illustrates that a single labeling is sufficient for use in classifier development.

Table 3.4 lists the inter-rater reliability of raters who labeled at least 8 meals. Overall the total agreements range from 89% to 98%. It should be noted that even for rater YS who labeled a large amount of gestures, the total agreement had 92% indicating a high degree of consistency for gesture definitions across the large data set.

Rater	#Gestures	Total agreement (%)	Agreement (%)	Boundary ambiguity (%)	Mistake (%)
YS	8481	7821 (92%)	6363 (75%)	1458 (17%)	660 (8%)
RB	942	864 (92%)	743 (79%)	121 (13%)	78 (8%)
AS	1107	983 (89%)	783 (71%)	200 (18%)	124 (11%)
JW	1182	1077 (91%)	937 (79%)	140 (12%)	105 (9%)
JP	818	800 (98%)	656 (80%)	144 (18%)	18 (2%)
PJ	649	619 (95%)	470 (72%)	149 (23%)	30 (5%)
TH	699	649 (93%)	499 (71%)	150 (21%)	50 (7%)
JD	1099	1012 (92%)	875 (80%)	137 (12%)	87 (8%)

Table 3.4: Inter-rater reliability for raters labeling at least 8 meals.

3.4 Conclusion

This chapter considers the problem of the lexicography of hand gestures during eating. Compared to the lexicography of hand gestures in sign language where the gesture vocabulary is designed top-down, the lexicography of hand gestures during eating must be designed bottom-up to encode a large variety of existing natural gesture behaviors. The goal of this chapter was to establish objective and repeatable definitions based on discernible intent during eating. A set of vocabulary of eating actions was built to quantify gestural behaviors. A total of 51,614 gestures were manually labeled and definitions were tested in a large data set for 276 participants. Duration of gestures varying from 1 second to 341 seconds indicates the difficulty of labeling segments. Inter-rater reliability of 18 raters showed 92.5% total agreement (75% exact agreement and 17.5% boundary ambiguity). The intake gestures had total agreement of 99.4% and 98.1% for bite and drink, respectively. Inter-rater reliability was further tested against a previously labeled data set of single time index-based labels. This test showed agreement of 94.4% and 95.8% for meals labeled by one and two raters, respectively. The performance of raters who labeled at least 8 meals was assessed, with total agreement ranging from 89% to 98%. Overall these findings show that the definitions are consistent and repeatable across a large data set.

Although the overall mistake rate is 7.5%, most mistakes are from non-intake gestures. By

design, a large variety of patterns resides in our existing definitions of non-intake gestures. For utensiling, stirring food and cutting food contain different patterns, where stirring involves more rotational motions while cutting involves periodic horizontal motions. Other actions such as peeling a fruit or vegetable or mixing food are also typical in utensiling. For rest, people have different levels of physiological tremor which is the natural variation in capability of holding perfectly still. Therefore our limited set of gesture labels has some difficulty in labeling all natural behaviors. Potential future work could explore an extension of our vocabulary to include additional gesture types or subdivide some gestures into multiple types. However, the purpose of this lexicography is to support research into the automated monitoring of dietary intake, where emphasis is on the detection and quantification of intake events. Recognizing a wider body of non-intake events may not be helpful towards this goal. Labeling a wider body of gesture types may also reduce inter-rater reliability.

A limitation of this work is that it was only tested on meals eaten in a cafeteria setting. It is possible that eating related gestures in other environments may require modifications to this encoding scheme. However, mitigating that problem is the fact that a relatively large number of people (276) were recorded eating a completely unscripted meal. Another limitation of this work is that it only considers gestures related to wrist motion. Other works [53, 71, 86, 100, 101] focus on eating actions related to the head or throat such as chewing and swallowing. This work could be extended to include lexicography for those types of events and should provide some background to assist with its development.

Chapter 4

The Impact of Quantity of Training Data on Recognition of Eating Gestures

Several studies have shown viable proofs-of-concept of recognizing eating gestures in laboratory settings with small numbers of subjects and food types [7, 97, 117, 129], but it is unclear how well these methods would work if tested on a larger population in natural settings. As more subjects, locations and foods are tested, a larger amount of motion variability could cause a decrease in recognition accuracy. This chapter explores the necessary complexity for a HMM to adequately capture the motion variability in eating gestures. We also test the effect of the quantity of training data needed to adequately train the HMMs. The work in this chapter has been submitted to the Journal of Biomedical and Health Informatics.

4.1 Introduction

Gesture recognition has been widely studied in the domain of sign language recognition [30, 130], motivating a similar approach for eating gesture recognition [97]. However, the variability in motion of an eating gesture is much larger than the variability in motion of a sign language gesture. Sign language gestures are specifically designed to communicate intent, and subject training

is conducted to minimize variability in repeated execution of the same gesture. In contrast, eating gestures are a result of a physiological activity (eating) and their execution varies depending on many variables including the subject, utensil, and food or beverage consumed.

For sign language recognition, one study found that an HMM constructed with 3 states and 3 Gaussians achieved 92% accuracy in recognizing 5,113 different words [30]. Another study found that 4 states with only 1 Gaussian achieved 91% accuracy in recognizing 25 different words [58]. For training data, one study found that 24 training samples per word achieved 95% accuracy for differentiating 300 different words [80]. Another study found that 60 training samples per word achieved 91% accuracy for differentiating 30 different words [4]. The general approach in all these works is to vary model complexity and/or quantity of training data to identify the point of diminishing returns in recognition accuracy. This chapter describes a similar effort for the recognition of eating gestures. The question is whether or not a similar model complexity (3-4 states, 1-3 Gaussians) and amount of training data (24-60 samples per word) are sufficient.

For eating gesture recognition, one study found that an HMM constructed with 5 states and 1 Gaussian achieved 94% in recognizing 384 gestures from 2 subjects [7]. Another study found that an HMM constructed with 13 states and 5 Gaussians achieved 84.3% in recognizing 2,786 gestures from 25 subjects [97]. For training data, one study found that 760 intake gestures from 10 subjects could train models that achieved 93% accuracy [129]. Another study found that 1,184 intake gestures from 14 subjects could train models that achieved up to 76% F1 score [117]. While the accuracies reported in these studies provide evidence of viable proofs-of-concept, it is unclear what accuracies would be achieved if the same methods were deployed on a larger population outside the laboratory.

The purpose of this chapter is to provide context on the relationship between a laboratory experiment that demonstrates a proof-of-concept, and a potential deployment of the same method on a larger population. Because of the inherent variability in eating gestures, one would expect that a classifier or algorithm trained on a small amount of laboratory data would not necessarily achieve the same accuracy when tested on a larger population. The experiments here provide some evidence on the size of training data and model complexity needed to provide confidence in that translation.

Study	#Subjects	#Gestures
[97]	25	2,786
[7]	2	384
[129]	10	760
[117]	14	1184
This work	269	51,614

Table 4.1: Statistics of data set in eating gestures.

4.2 Methods

4.2.1 Data Collection

The data set from Chapter 3 is used here, where a total of 276 subjects were recruited and each consumed a single meal. For 5 subjects, either the video or wrist motion tracking failed to record, and for 2 subjects non-dominant hands were used for recording; these a excluded from analysis.

4.2.2 Data Preprocessing

Let the data set be comprised of d eating sessions, where each session is a recording of wrist motion during contiguous consumption. For example, a meal might be divided into multiple eating sessions (appetizer, entree, dessert) separated by periods of non-eating that are not recorded. The wrist motion data is defined as $R_t^d = [x_t^d, y_t^d, z_t^d, \alpha_t^d, \beta_t^d, \gamma_t^d]$, where d indicates the eating session, t represents the time index, x, y, z are accelerometer sensor readings and α, β, γ are gyroscope sensor readings. All the data were smoothed using a Gaussian-weighted window of width 1 s and standard deviation of $\frac{2}{3}$ s:

$$\tilde{R}_t^d = \sum_{i=-N}^{0} R_{t+i}^d \frac{exp\left(\frac{-t^2}{2\sigma^2}\right)}{\sum_{x=0}^{N} exp\left(\frac{-(x-N)^2}{2\sigma^2}\right)} \tag{4.1}$$

4.2.3 Hidden Markov models

Figure 4.1 illustrates the architecture of our HMMs. There are two levels, HMM-S and HMM-N. The first level observes the motion subcomponents of a single gesture and classifies the motion sequence according to which gesture it most resembles. The second level observes the probable identities of a preceding set of gestures and classifies the current gesture according to which

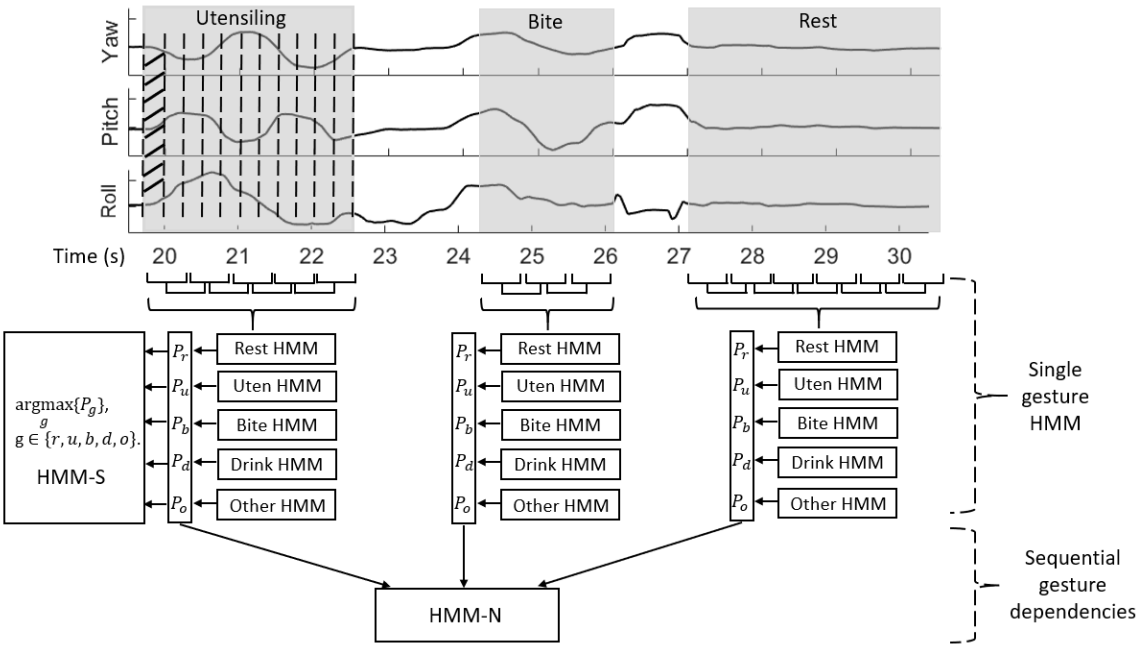


Figure 4.1: Architecture for single gesture HMM-S and gesture-to-gesture HMM-N. Examples of three manually segmented gestures are displayed. In HMM-S, the observables are a sequence of features computed from the raw sensor data (only gyroscope signals are shown for brevity) in sliding windows, each with 50% overlap denoted by the shaded area. Each gesture type (rest, uten., etc.) is recognized using a different HMM. For each input sequence, the HMM with the maximum logarithmic probability determines the gesture type. Gesture sequence recognition uses the set of logarithmic probabilities as observables for HMM-N, in which each state represents a sequence of N gestures.

gesture sequence it most resembles. The details of each level of HMM are as follows.

4.2.3.1 Single Gesture HMM-S

We use HMM-S to model a single gesture as a sequence of sub-gestures with each sub-gesture represented by a state [97]. For example, the action of taking a bite may consist of raising food towards the mouth, ingestion, and the return of the wrist to a rest position. This sequence is modeled through a state sequence where each state models part of the motion pattern.

We use the notation $\lambda = (\pi, A, B)$ for each HMM, where A, B and π are the state transition matrix, emission probability and initial state distribution, respectively. We denote individual states as $S = \{s_1, s_2, ..., s_N\}$, the state at time t as q_t, and a state sequence as $Q = \{q_1 q_2, ..., q_T\}$. The initial state distribution π is computed as:

$$\pi_i = P(q_1 = s_i), \quad 1 \leq i \leq N. \tag{4.2}$$

The state transition matrix A is computed as:

$$a_{ij} = P(q_{t+1} = s_j | q_t = s_i), \quad 1 \leq i, j \leq N, 1 \leq t \leq T. \tag{4.3}$$

The observables O for each gesture g are calculated using two windows w_1 and w_2, where w_1 is the length of time in which features are calculated and w_2 is the step in time between feature calculations. Formally, we calculate features as in Equations 4.4-4.6:

$$\mu_{g,t} = \frac{1}{w_1} \sum_{i=0}^{w_1} \tilde{R}_{g,t+i} \tag{4.4}$$

$$\sigma_{g,t} = \sqrt{\frac{1}{w_1 - 1} \sum_{i=0}^{w_1} (\tilde{R}_{g,t+i} - \mu_{g,\tilde{R},t})^2} \tag{4.5}$$

$$s_{g,t} = \frac{(\tilde{R}_{g,t+w_1} - \tilde{R}_{g,t})}{w_1} \tag{4.6}$$

where $\tilde{R}_{g,t}$ is the smoothed sensor reading of gesture g at time t, and $\mu_{g,t}$, $\sigma_{g,t}$, $s_{g,t}$ are the average, standard deviation and slope. In each window w_1, this provides 18 features $o_{g,t} = [\mu_{x_{g,t}}, \mu_{y_{g,t}}, ..., \sigma_{x_{g,t}}, \sigma_{y_{g,t}}, ..., s_{x_{g,t}}, s_{y_{g,t}}, ..., s_{\gamma_{g,t}}]$. The features for each gesture g can be represented as $O_g =$

$[o_{g,t}, o_{g,t+w_2}, ..., o_{g,t+l\times w_2}]$, where l depends on the duration of each gesture.

We select w_1 and w_2 to act as a sliding window with overlap. The classic approach in speech recognition is to use a 0.5 second window with 50% overlap [65]. Since our data is collected at 15 Hz, we use the odd number of a 9 sample window (0.6 s) with 4 sample overlap (0.3 s). We perform a z-score independently for each of the 18 features to prevent skew towards large valued features.

Gaussian mixture models (GMMs) are used to describe the emission probabilities B, as shown in Equation 4.7, where M is the number of Gaussians.

$$B = P(O|Q) = \sum_{i=1}^{M} c_i N(O; \mu_i, \Sigma_i), \quad \sum_{i=1}^{M} c_i = 1. \tag{4.7}$$

Each Gaussian is defined by three parameters c_i, μ_i, and Σ_i representing weight, mean and covariance matrix of the i^{th} Gaussian, respectively:

$$N(O; \mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(O-\mu)^T \Sigma^{-1}(O-\mu)} \tag{4.8}$$

We assume features are independent and the off-diagonal entries in Σ are zero. The expectation-Maximization algorithm is used to calculate the emission probabilities modeled by GMMs [36, 61].

An HMM toolbox was used to build HMMs [81]. We use an architecture of left-to-right with skip in HMM-S [106], so π is always one for the first state and zero for the other states. The forward-backward algorithm is used to train an HMM, in other words to learn the A and B matrices given an observation sequence O.

In HMM-S, five HMMs are built, one for each gesture type. During recognition, observables O from an unknown gesture are passed into the HMMs as illustrated in Figure 4.1. Each HMM λ_g computes the likelihood of this particular observation sequence using the forward algorithm:

$$P(O|\lambda_g) = \sum_{Q} P(O, Q|\lambda_g) = \sum_{Q} P(O|Q, \lambda_g) P(Q|\lambda_g). \tag{4.9}$$

Each unknown gesture obtains probability scores from each of the five HMMs. A probability score indicates how well a model matches the gesture. Therefore, the model which provides the maximum

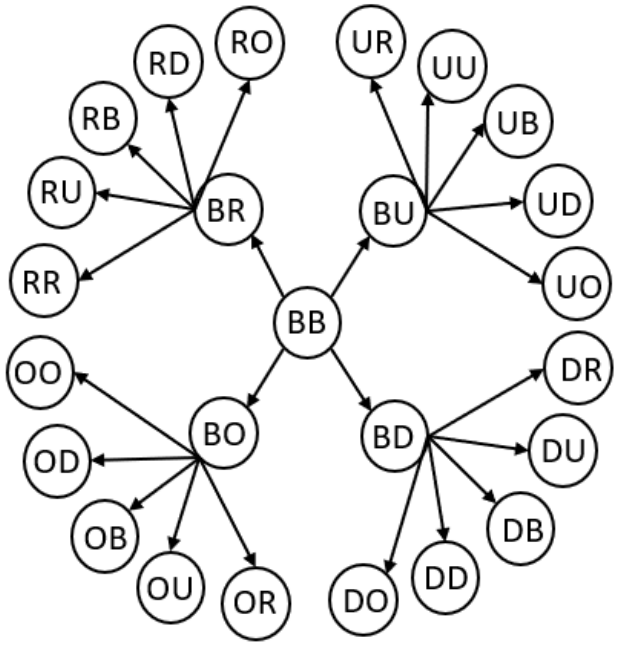


Figure 4.2: State transitions in HMM-2. For clarity purpose, 25 transitions starting from bite is displayed. B = bite, D = drink, R = rest, U = utensiling, O = other.

score determines the gesture type g ∈ {rest, utensiling, bite, drink, other} as

$$\hat{g} = \arg\max_{g}\{P_g\} \tag{4.10}$$

4.2.3.2 Sequential Dependent HMM-N

We use HMM-N to model a sequence of N previous gestures as context to improve recognition of the current gesture. For example, a common pattern is to use utensils to prepare a bite of food (U), consume the bite of food (B), and then rest hands while masticating and swallowing (R). In HMM-N, each state models a sequence of N gestures. Figure 4.1 illustrates the architecture of HMM-N. Note that the observables and states in HMM-N are different from those used for HMM-S.

To calculate the state transition matrix, we convert HMM-N to an equivalent first-order HMM [74, 97]. Figure 4.2 shows a partial example of the equivalent first-order HMM for HMM-2 (for clarity, only state transitions starting from bite are shown). Logically, transitions between some states are impossible. For example, state BB cannot transition to DB because the former state's

most recently recognized gesture is B, which does not match the memory of the latter, which is D. In total HMM-2 has only 5×25 possible state transitions. Formally, each state in HMM-N is $s_i = \{g_1 g_2 ... g_N\}$. The state transition matrix A is calculated as:

$$\begin{aligned} a_{ij} &= P(s_j = g_2 g_3 ... g_{N+1} | s_i = g_1 g_2 ... g_N) \\ &= \frac{\#\text{transitions from } g_1 g_2 ... g_N \text{ to } g_2 g_3 ... g_{N+1} + 1}{\#\, g_1 g_2 ... g_N \text{ gesture sequences} + |S|} \end{aligned} \tag{4.11}$$

where $|S|$ indicates the number of possible state transitions. Laplace smoothing (+1 in numerator, $+|S|$ in denominator) is used to avoid values of zero in the state transition matrix (cases in which a sequence does not appear in the training data) [104].

The initial state distribution π is calculated as:

$$\begin{aligned} \pi_i &= P(s_i = g_1 g_2 ... g_N) \\ &= \frac{\#\, g_1 g_2 ... g_N \text{ gesture sequences} + 1}{\#\,\text{N-gesture sequences} + |S|} \end{aligned} \tag{4.12}$$

The observables O of each gesture are the five probability scores from HMM-S. In HMM-N, we make an assumption that the observables are only dependent on the most recent gesture. For example, observables from gesture D, gesture sequence UD and UUD are the same (they are all immediate observations of D). The emission probabilities B are calculated as:

$$\begin{aligned} B &= P(O | s_i = g_1 g_2 ... g_N) \\ &= P(O | g_N) = \sum_{i=1}^{M} c_i N(O; \mu_i, \Sigma_i). \end{aligned} \tag{4.13}$$

We follow the same parameter setting in [97] to use 7 Gaussians.

During training, A and π are calculated using Equations 4.11-4.12 and B is learned using Equation 4.13. Recognition is defined as finding the most likely state sequence $Q = \{q_1, q_2, ...q_T\}$ that explains observables O given model $\lambda = \{A, B, \pi\}$:

$$Q = \underset{q_1, q_2, ..., q_T}{\arg\max}\ P(Q | O, \lambda) \tag{4.14}$$

The Viterbi algorithm [61] is used and the most recent gesture in each q_t determines the gesture

type for each time step.

4.2.4 Model Complexity and Training Data

To study the amount of motion variability within each gesture type, we varied the number of states N and the number of Gaussians M for HMM-S. Specifically we built every combination of HMM-S with $N = 3...25$ and $M = 1...7$. The value for N can be considered to correspond to the number of different sub-motions expected in a gesture type. The value for M can be considered to correspond to the number of observed variations of each expected sub-motion. During training, we randomly selected 650 gestures of each type to train the 5 HMMs. During recognition, we selected another set of 650 gestures per type, excluding those used in training, to test the accuracy. Due to the Monte Carlo nature of HMM training, each model was run 5 times and the average is reported.

To study the effect of the quantity of training data, we varied the amount of gesture samples used to train each HMM. The values for N and M were held constant according to the best values found from the model complexity experiment. We varied the number of training samples per gesture type from 65 to 650 by randomly selecting from the full data set. During recognition, the same set of testing data as above was used. To reduce the variance introduced in the process of random selection of training data, each model was run 30 times and the average is reported.

4.3 Results

The total data set consists of 51,614 manually labeled gestures, with 14,761 rest, 14,861 utensiling, 18,462 bite, 2,182 drink and 1,348 other. This data set, along with a visualization tool, is being made publicly available at `http://www.cecas.clemson.edu/tracking`.

4.3.1 HMM-S

Figure 4.3 shows recognition accuracy vs HMM complexity. Accuracy plateaus at $M = 5$ and $N = 13$, indicating that 5 Gaussians and 13 states are needed to capture the motion variability.

Figure 4.4 shows recognition accuracy vs number of training gestures, with model complexity fixed at $M = 5$ and $N = 13$. Accuracy plateaus at 500 training samples per gesture type, indicating that while 65 training samples per gesture type are the minimum needed to train HMMs of this complexity, an additional 8% accuracy is achieved by training with 500 samples per gesture type.

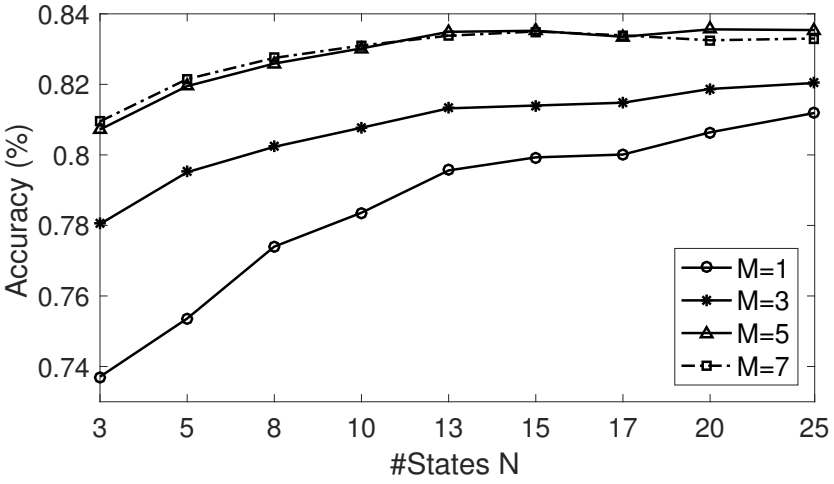


Figure 4.3: Recognition accuracy with model complexity: the number of states N and mixture components M.

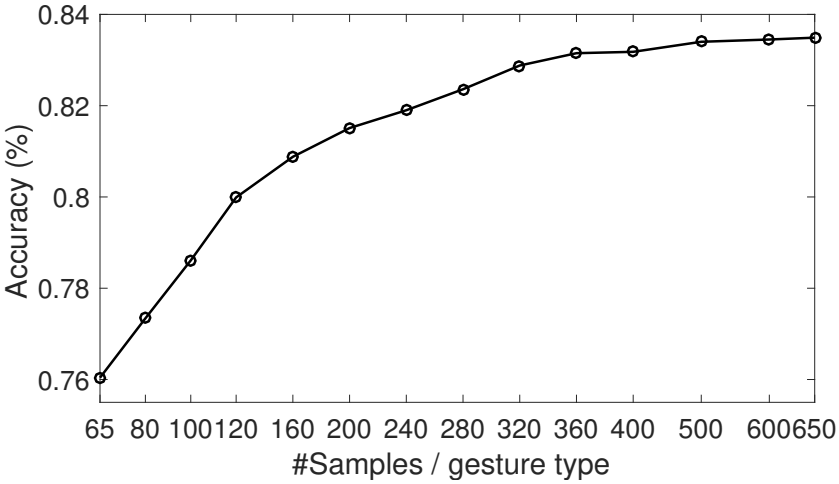


Figure 4.4: Recognition accuracy with the quantity of training data.

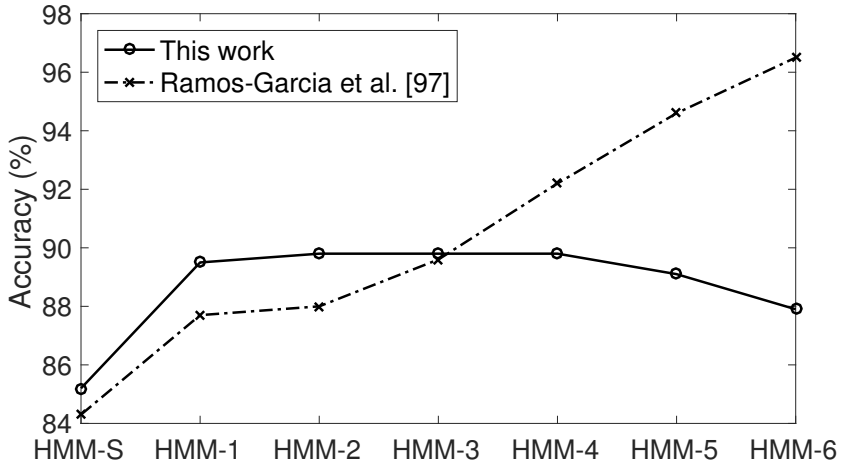


Figure 4.5: Accuracy of models trained on different amount of data.

4.3.2 HMM-N

Figure 4.5 shows the accuracy for HMM-N as different amounts of previous gestures (S=0, N=1...6) are incorporated into the classifier. The figure compares our results for 51,614 gestures for training vs the same model trained on a smaller data set of 2,786 gestures from 25 meals [97]. The accuracy of HMM-1 is improved by 1.8% by having more training data, although part of this can be explained by the simultaneous 0.9% increase in accuracy in HMM-S that produces the inputs used for HMM-1. However, the accuracies for HMM-2 and higher are not improved by more training data. One reason is that building HMM-N requires converting high-order HMMs to first order by enumerating every possible combination between current and previous gestures. This requires a large amount of parameters in states and transition matrices, as shown in Table 4.2. In HMM-N, the number of states is 5^N and possible state transitions is 5×5^N. Emission probabilities are modeled by mean μ and covariance Σ of GMMs, as shown in Equation 4.13. Given M mixture components and D-dimensional observables, the amount of emission probability parameters are $5 \times D \times M$ for μ and Σ, respectively. Based on the one in ten rule in building models [40, 41, 91], it is necessary to build models with a quantity of training data at least 10 times the number of parameters. Therefore, the inadequate training data in [97] caused overfitting for models from HMM-3 to HMM-6. It is worth noting that even with the large data set of 51,614 gestures, we still do not have adequate data to train HMM-5 and HMM-6. Another reason is that the transition matrix of testing data in [97] was included during training to avoid cases where gesture sequence in training data does not exist

#Params.	HMM-1	HMM-2	HMM-3	HMM-4	HMM-5	HMM-6
Prior	5	25	125	625	3,125	15,625
Transition	25	125	625	3,125	15,625	78,125
Emission	350	350	350	350	350	350
Total	380	500	1,100	4,100	19,100	94,100

Table 4.2: #Parameters in HMM-N. Note: observable is 5-dimensional vector and 7 GMMs are used.

Model	All (%)	Rest (%)	Utensiling (%)	Bite (%)	Drink (%)	Other (%)
HMM-S	85.2	86.8	86.9	83.7	96.1	52.6
HMM-1	89.5	89.0	91.2	91.9	91.2	52.2

Table 4.3: Recognition accuracy for HMM-S and HMM-1.

in testing data, but this biased the results. Here we used Laplace smoothing instead.

Finally, Table 4.3 summarizes the accuracies of HMM-S and HMM-1 trained and tested on all data using five-fold cross validation. Overall, HMM-S achieves 85.2% accuracy and HMM-1 achieves 89.5% accuracy. For each gesture, the improved accuracy for rest, utensiling and bite is 2.2%, 4.3%, 8.2%, respectively. We observe that drink and other decrease in accuracy from HMM-S to HMM-1, suggesting that there is not enough sequencing consistency in their occurrence in a large data set to warrant modeling their sequencing in an HMM.

4.4 Discussion

In this chapter, two models were built: HMM-S which models the sequence of actions within a gesture, and HMM-N which models the sequential dependence between gestures. A total of 51,614 gestures with 5 different gesture types were labeled from 269 subjects eating a single meal in a cafeteria environment. Sign language HMMs constructed with only 3-4 states and 1-3 Gaussians achieved more than 90% accuracy [30, 58], whereas we found that eating gesture HMMs needed 13 states and 5 Gaussians to achieved 85.2% accuracy. For training data, in contrast to sign language in which only 24-60 training samples per word were sufficient, we found that 500 training samples per gesture type were required. These findings demonstrate that the variability of motion patterns in eating gestures are much larger than the variability in motion patterns in sign language, and that more complex models and more training data are required. For HMM-N, the effect of model complexity was explored by studying the sequential dependence of N previous

gestures to improve recognition of the current gesture. Results show that accuracy was improved by 4.3% when one previous gestures was studied as context, but was not improved when additional previous gestures were studied. This demonstrates that word/gesture sequencing, commonly used to improve speech/sign language recognition accuracy, may have less applicability to eating gesture recognition.

One of the key challenges for eating gesture recognition is to translate models built with laboratory data to models built with free-living data. Most previous works have trained models with data in a limited amount of subjects, time collected and laboratory environment [7, 117, 129]. However, a large variability exists in eating gestures during free-living and models that are developed in a controlled setting will potentially be brittle in a natural setting. For example, people typically gesticulate when talking to others or place a phone call while eating, which do not happen in a laboratory environment. Recently, several studies have been investigating the differences between performance in laboratory and real-world environments. Study in [118] trained models using 10 hours of data collected in laboratory from 20 subjects and tested on 31 hours of data collected in free-living from 7 subjects, and 422 hours of data collected in free-living from 1 subject, achieved 76% and 71% F-score, respectively. Another study [79] trained models using 59 hours of data collected in the laboratory from 6 subjects and tested on 113 hours of data collected in free-living from the same group of subjects, achieved 88% precision and 87% recall. A third study [132] detected chewing events on 122 hours of data collected in free-living from 10 subjects, achieved 79% recall and 77% precision, respectively. However, it is difficult to directly compare with our work. First, different sensing modalities are investigated, e.g. sensing of wrist motion, chewing and swallowing. Second, while these works focus only on eating detection, our work also recognizes some eating related activities, such as utensiling. Finally, the purpose of our work is to explore the quantity of training data for the model to adequately capture the motion variability in a large population.

One limitation is that data was collected only one meal per subject, which might not be adequate to capture variability within individuals. Another limitation is that data was collected only in a cafeteria location, which might not capture the full motion variability of free-living eating behaviors. In future work we would like to collect free-living data from individuals over a longer duration (e.g., a week or more) and in multiple locations.

Chapter 5

Recognizing Eating Gestures Using Contextual Dependent Hidden Markov Models

5.1 Introduction

This chapter considers the problem of recognizing eating gestures by studying contextual variables using HMMs to capture motion variations. Specifically, we examined if foreknowledge of the demographics (gender, age, ethnicity, BMI, handedness), meal level variables (utensil being used, types of foods being eaten), language variations (variations of bite, utensiling and other), and clustering based method could improve recognition accuracy. Improvement in accuracy was measured by comparing to the non-HMM baseline algorithm in Chapter 2, as well as HMM-S and HMM-N proposed in Chapter 4.

5.2 Methods

5.2.1 Data

The data set from Chapter 3 is used here. We provide the demographics and food information for the clarity of the contextual variables. Participants were free to choose any foods and beverages available. In total, 380 different food and beverage types were chosen, for example stir fry vegetables, pasta, shoestring French fries, salad bar, water, soda, etc. Four different utensils were used: fork, spoon, chopsticks and hand. Tables 5.1-5.5 list the gender, age, hand used, ethnicity, and BMI distributions of participants that were tested as contextual variables in this work.

Gender	#Participants
Male	129
Female	140

Table 5.1: Gender distribution of participants.

Age	#Participants
18-30	188
31-40	27
41-50	33
51-75	21

Table 5.2: Age distribution of participants.

Hand used during eating	#Participants
Right	248
Left	21

Table 5.3: Hand used distribution of participants.

5.2.2 Context Dependent HMMs

To study the contextual variables, two methods have been investigated: top-down and bottom-up. In the top-down approach, contextual variables are defined first and then used to capture motion variations. For example, the demographics, meal-level variables, language variables are the top-down contextual variables. In the bottom-up approach, contextual variables are unknown and determined based on the motions participants take during eating.

Ethnicity	#Participants
African American	26
Asian or Pac. Isl.	29
Caucasian	188
Hispanic	11
Other	15

Table 5.4: Ethnicity distribution of participants.

BMI	#Participants
< 25	164
25-30	66
>=30	39

Table 5.5: BMI distribution of participants.

5.2.2.1 Top-down approach

In the top-down approach, demographics of participants (gender, age, ethnicity, BMI, handedness), meal-level variables (utensil being used, types of foods being eaten) and language variables (bite sub-vocabulary, utensil sub-vocabulary and other sub-vocabulary) were studied to determine if they provide increased recognition accuracy compared to HMM-S. Each contextual variable was test independently.

Demographics HMMs

Gender HMMs. HMMs were trained independently for females and males using the same steps described for the HMM-S. This yielded 10 total HMMs (2 genders $\times$ 5 gesture types). During the recognition process, it is assumed that the gender of the participant is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. Figure 5.1 shows the process. After selecting gender, observbles of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

Age HMMs. HMMs were trained independently for each age category listed in Table 5.2. This yielded 20 total HMMs (4 age groups $\times$ 5 gesture types). During the recognition process, it is assumed that the age of the participant is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting age group, observbles of the unknown gesture are passed into 5 HMMs

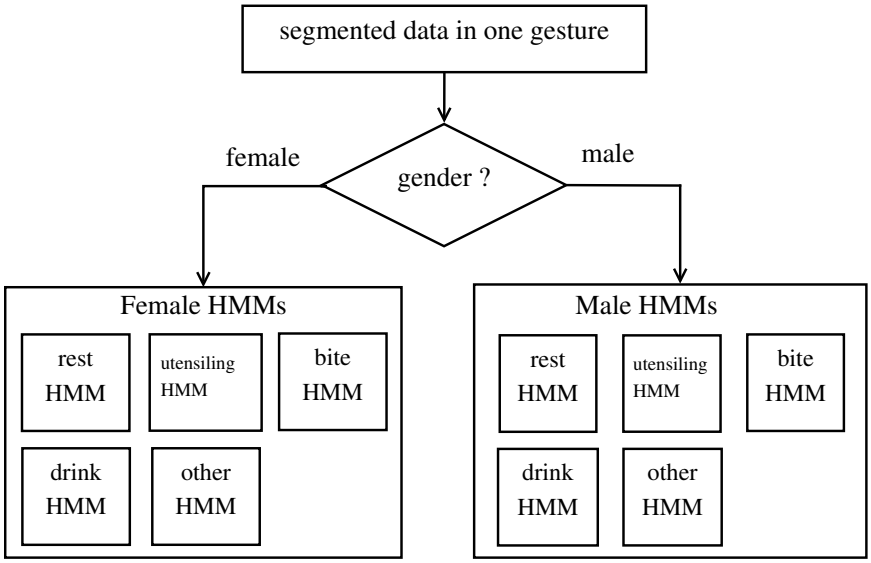


Figure 5.1: Gender HMMs gesture recognition.

and the HMM which provides the maximum score determines the gesture type.

Hand HMMs. HMMs were trained independently for each category listed in Table 5.3. This yielded 10 total HMMs (2 hand groups $\times$ 5 gesture types). During the recognition process, it is assumed that the hand used by the participant is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting hand group, observables of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

Ethnicity HMMs. HMMs were trained independently for each ethnicity category listed in Table 5.4. This yielded 25 total HMMs (5 ethnicity groups $\times$ 5 gesture types). During the recognition process, it is assumed that the ethnicity of the participant is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting ethnicity group, observables of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

BMI HMMs. HMMs were trained independently for each BMI category listed in Table 5.5. This

Utensil	#Meals
Fork	153
Spoon	47
Hands	149
Chopsticks	5
Mixed	134

Table 5.6: Utensil distribution of meals.

yielded 15 total HMMs (3 BMI groups $\times$ 5 gesture types). During the recognition process, it is assumed that the BMI of the participant is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting BMI group, observables of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

Meal level HMMs

We investigated the utensil used and the food consumed in each meal to build sub-gesture HMMs. In each meal, the dominant utensil used and the food consumed was determined beforehand. If there were no dominant utensil types or food, then the type of that meal was determined as mixed.

Utensil HMMs. Four utensil types were available in our data set: fork, spoon, hands and chopsticks. However, utensil use is not necessarily unique throughout an entire meal. For example, a participant may use a fork for some bites and hands for other bites. Therefore, we defined a fifth category as mixed utensil. If no single utensil type was used for more than 65% of bite gestures by a participant, then their utensil type was considered mixed. Table 5.6 lists the totals.

This yielded 25 total HMMs (5 utensil types $\times$ 5 gesture types). During the recognition process, it is assumed that the utensil type of each meal is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting utensil type, observables of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

Food HMMs. In total 380 foods were available in our data set. Table 5.8 lists the frequencies of food and beverage of which participants consumed greater than 100 bites. In each meal, one or multiple food types may be consumed. Therefore, we determined the dominant food type if the

Food accuracy	#Meals
High accuracy	137
Middle accuracy	66
Low accuracy	16
Mixed	269

Table 5.7: Food distribution of meals.

frequency of food was consumed more than 55% within each meal. If there was no dominant food, then it was determined as mixed. Based on the accuracy of detecting food using the algorithm in Chapter 2, four groups were generated: food accuracy larger than 75% (high accuracy), between 60% and 75% (middle accuracy), less than 60% (low accuracy) and mixed food type. Table 5.7 lists the totals.

This yielded 20 total HMMs (4 food types $\times$ 5 gesture types). During the recognition process, it is assumed that the food type of each meal is known a priori and thus can be used to determine which set of HMMs to use to recognize gestures. This selection process is similar to the one outlined in Figure 5.1. After selecting food type, observables of the unknown gesture are passed into 5 HMMs and the HMM which provides the maximum score determines the gesture type.

Language variations HMMs

Originally, five gestures were defined to capture eating activities. However, a large variety of patterns resides in the existing gestures. For utensiling, stirring food and cutting food contain different patterns, where stirring involves more rotational motions while cutting involves periodic horizontal motions. Other actions such as peeling a fruit or vegetable or mixing food are also typical in utensiling. For bite, different utensils can be used to perform: fork, spoon, hands or chopsticks. Therefore in this section, we investigated extending the current vocabularies of bite, utensiling and other to capture detailed information.

Bite variational HMMs. Based on the utensils used for each bite, we divided bite gestures into five variations: bite with fork, bite with spoon, bite with both hands, bite with single hand and bite with chopsticks. Table 5.9 lists the amount of each gesture.

This yielded 9 total HMMs. During the recognition process, each gesture was passed into 9 HMMs, with gesture type determined by the model with max score. This selection process is

Food	#Frequency
Salad_bar	3986
Shoe_string_french_fries	1791
Water	976
Pasta_tour_of_italy	756
Sweet_tea	714
Ice_cream	484
Stir_fry	426
Pepperoni_pizza	377
Ice_cream_cone	314
Cheese_pizza	273
Diet_coke	246
Signature_chips	218
Cantaloupe	214
Homestyle_chicken_sandwich	205
Pad_thai_shrimp_station	195
Veggie_pizza	183
Hamburger	181
Custom_sandwich	171
Cherry_coke	168
Coca_cola	165
Apple	165
Hunan_chicken	160
Bread	160
Cereal_apple_jackse	157
Baked_honey_bbq_lemon_chicken	151
Apple_juice	151
Oven_fried_chicken	149
Frozen_yogurt	148
Garlic_breadsticks	146
Baked_rotisserie_chicken	140
Fried_shrimp	138
Yogurt	132
Chicken_caesar_wrap	131
Pita_bread	127
Coke_zero	121
Grapefruit	119
Kiwi_strawberry_juice	117
Waffle_bar	114
Pork_chop_suey_with_white_rice	110
Hunan_chicken_and_rice	106
Brownie	105
Lemonade	103

Table 5.8: Frequency for foods of which participants consumed greater than 100 bites.

Gesture type	#Gestures
Rest	14761
Utensiling	14861
Drink	2182
Other	1348
Bite-fork	9724
Bite-spoon	2507
Bite-both hands	1376
Bite-single hand	4291
Bite-chopsticks	564

Table 5.9: The number of gestures of bite variations.

Gesture type	#Gestures
Rest	14761
Bite	18462
Drink	2182
Other	1348
Utensiling-fork	7416
Utensiling-spoon	1585
Utensiling-both hands	302
Utensiling-single hand	1402
Utensiling-chopsticks	294
Utensiling-other	3862

Table 5.10: The number of gestures in utensiling variations.

outlined in Figure 5.2. Notice that if the gesture was recognized as one of the five bite variations, it was then determined as "bite".

Utensiling variational HMMs. In total there are 5 utensil types in our data set: fork, spoon, both hands, single hand and chopsticks. The variations of the gesture utensiling is categorized based on the following gesture: if the following gesture is bite, then the utensiling type is determined by the utensil used for that bite, otherwise it is categorized as "utensil-other". As a result, utensiling variations include: utensiling-fork, utensiling-spoon, utensiling-both hands, utensiling-single hand, utensiling-chopsticks and utensiling-other. Table 5.10 lists the amount of each gesture.

This yielded 10 total HMMs. During the recognition process, each gesture was passed into 10 HMMs, with the gesture typed determined by the model with max score. This selection process is similar in Figure 5.2. Notice that if the gesture was recognized as one of the six utensiling variations, it was then determined as "utensiling".

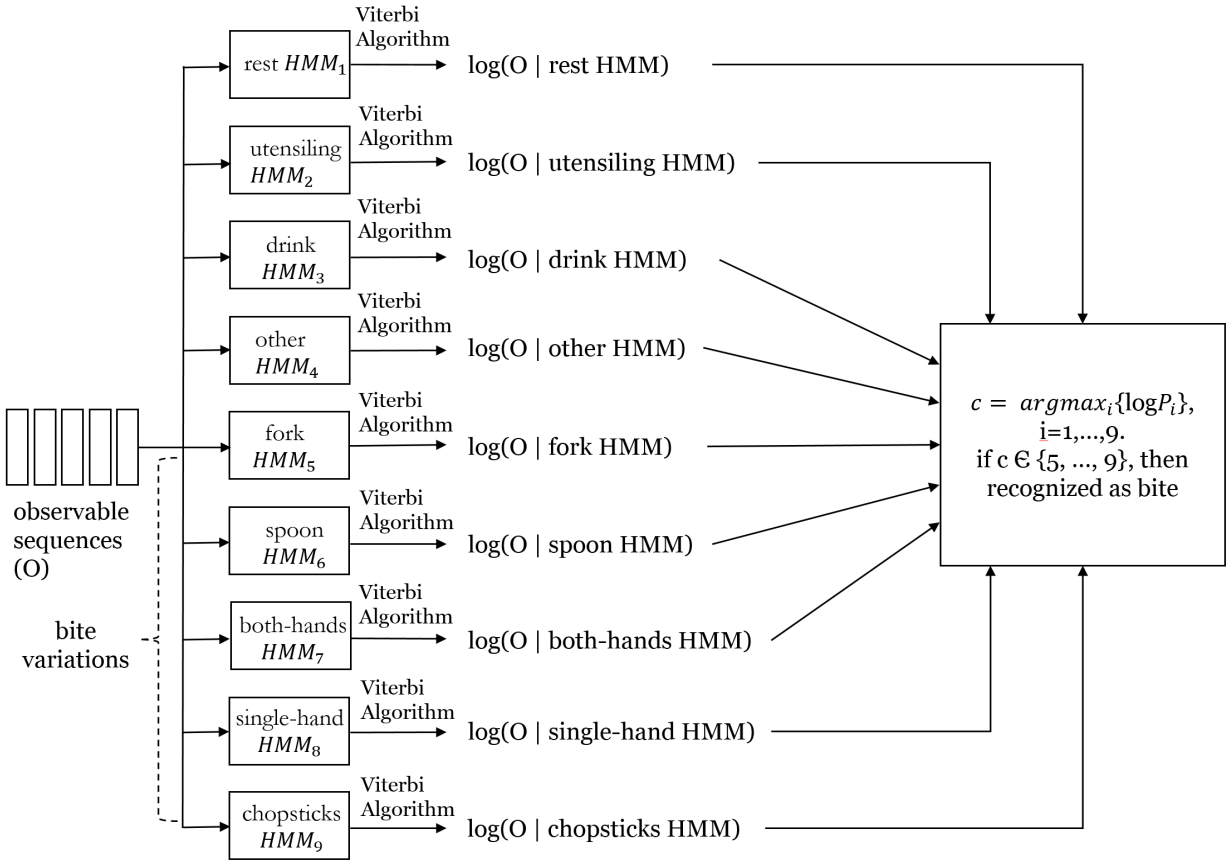
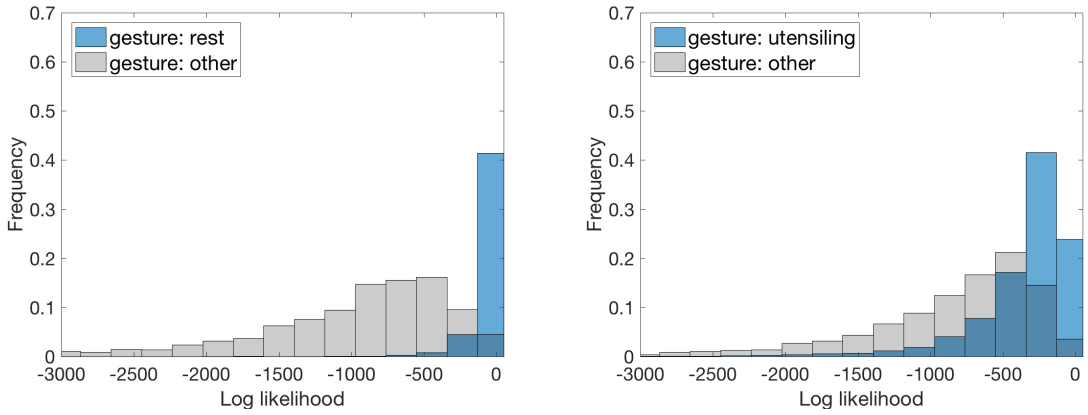
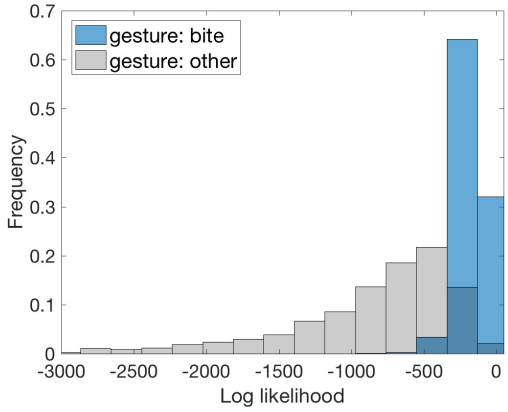


Figure 5.2: Recognition of bite variational HMMs.

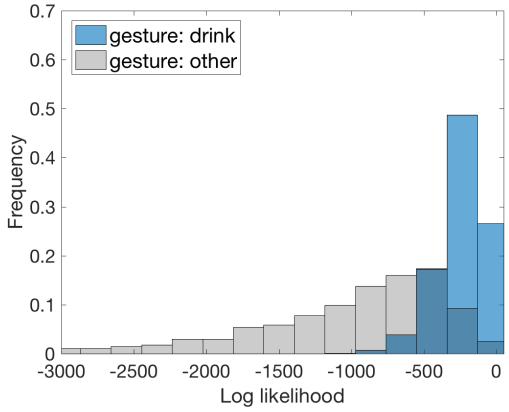


(a) Log scores from rest model.

(b) Log scores from utensiling model.



(c) Log scores from bite model.



(d) Log scores from drink model.

Figure 5.3: Log scores of other compared with rest, utensiling, bite and drink.

Other variational HMMs. A lot of variations exist in gesture "other" based on its definition. For example, other can include using napkin, gesturing while talking, playing the mobile phone, etc. Some gestures of other share similar motions with rest, utensiling, bite and drink. To reduce the ambiguity, we investigate dividing other based on its log score from HMMs: if the log score of other is very different from the scores of the rest 4 gestures, then it is considered as dissimilar with any gestures, otherwise it has some similar motions and can be misclassified. Figure 5.3 displays the log scores of other compared with 4 gestures. For example, in Figure 5.3a, the scores of other and rest are computed from the model "rest" and are compared. Other with log scores under -500 are considered as dissimilar with rest. Others with log scores under -500, -1500, -500 and -1000 from rest model, utensiling model, bite model and drink model are considered as dissimilar with all 4 gestures, otherwise categorized as similar with any 4 gestures. Table 5.11 lists the total.

Gesture type	#Gestures
Rest	14761
Utensiling	14861
Bite	18462
Drink	2182
Other-dissimilar with all four gestures	404
Other-similar with any gestures	944

Table 5.11: The number of gestures in other variations.

This yielded 6 total HMMs. During the recognition process, each gesture was passed into 6 HMMs, with the gesture typed determined by the model with max score. This selection process is similar in Figure 5.2. Notice that if the gesture was recognized as one of the two other variations, it was then determined as "other".

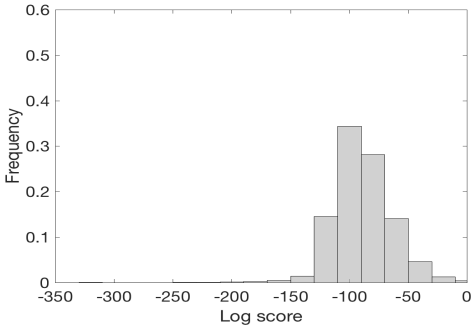
5.2.2.2 Bottom-up approach

In the bottom-up approach, contextual variables are determined by a clustering method to capture motion variations. Specifically, we investigate extending the sub-vocabularies of bite by kmeans and log probability scores from HMM-S to improve recognition accuracy.

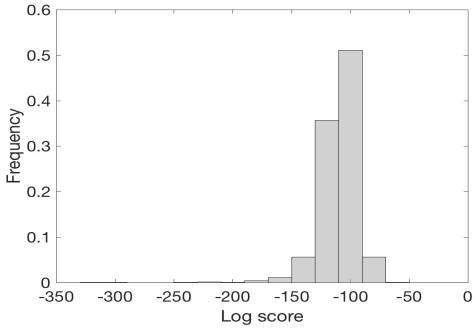
Log-score clustered HMMs. In log-score clustered HMMs, gesture "bite" is grouped into eight clusters based on the log scores from HMMs. Specifically, the HMM for bite is built and the log scores of each bite is computed. Log score indicates how well HMM captures the bite motions, where the higher the score the better the model characterizes that bite. Then bites are divided into two groups, where one group contains bites with top 50% score and the other group contains bites with bottom 50% score. In total the same procedure is performed three times and eight bite clusters are generated. Figure 5.4 displays the log score distribution of eight bite clusters. Table 5.12 lists the amount of each gesture.

This yielded 12 total HMMs. During the recognition process, each gesture was passed into 12 HMMs, with gesture type determined by the model with max score. This selection process is similar in Figure 5.2. Notice that if the gesture was recognized as one of the eight bite clusters, it was then determined as "bite".

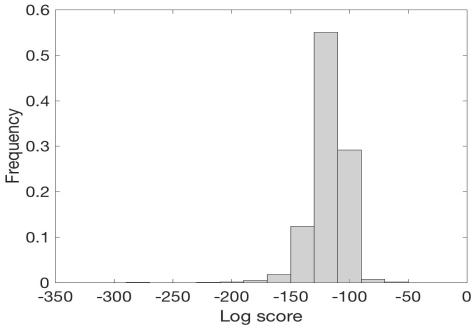
Kmeans clustered HMMs. In kmeans clustered HMMs, bite clusters are generated by kmeans



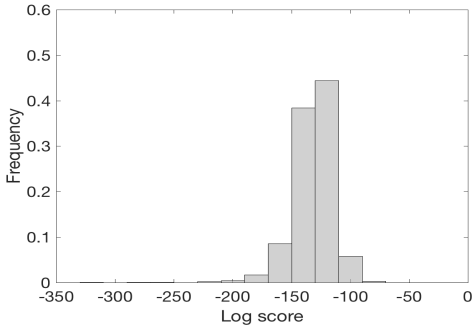
(a) Log score distribution from 1st bite cluster.



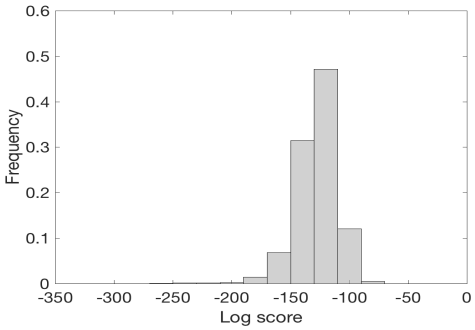
(b) Log score distribution from 2nd bite cluster.



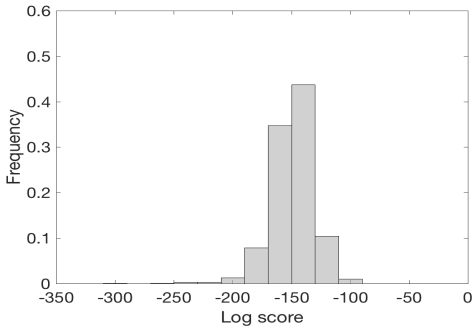
(c) Log score distribution from 3rd bite cluster.



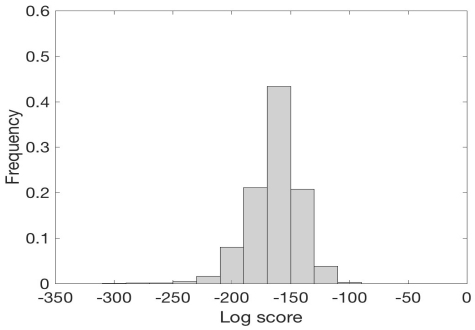
(d) Log score distribution from 4th bite cluster.



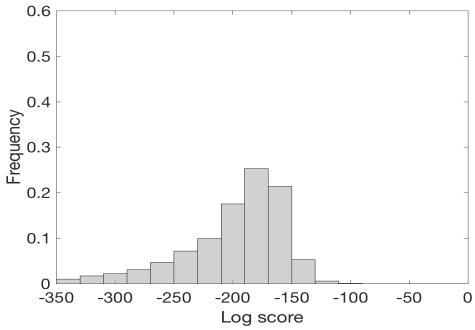
(e) Log score distribution from 5th bite cluster.



(f) Log score distribution from 6th bite cluster.



(g) Log score distribution from 7th bite cluster.



(h) Log score distribution from 8th bite cluster.

Figure 5.4: Log scores of eight bite cluster HMMs.

Gesture type	#Gestures
Rest	14761
Utensiling	14861
Drink	2182
Other	1348
Bite-1st cluster	2309
Bite-2nd cluster	2308
Bite-3rd cluster	2308
Bite-4th cluster	2307
Bite-5th cluster	2309
Bite-6th cluster	2307
Bite-7th cluster	2308
Bite-8th cluster	2306

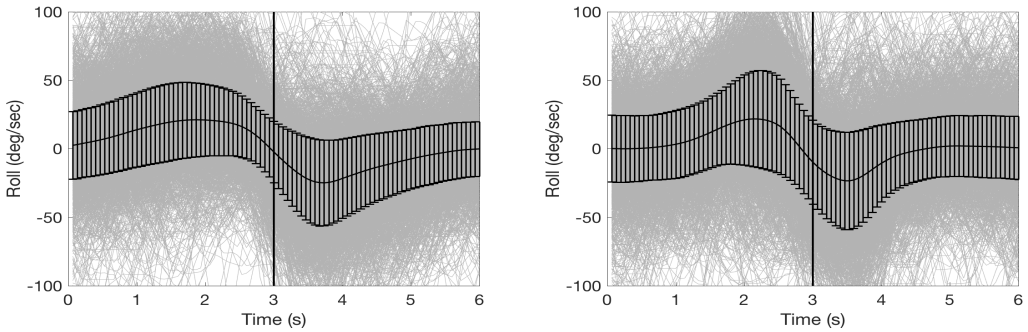
Table 5.12: The number of gestures in log-score clustered approach.

Gesture type	#Gestures
Rest	14761
Utensiling	14861
Drink	2182
Other	1348
Bite-1st cluster	4101
Bite-2nd cluster	3756
Bite-3rd cluster	2361
Bite-4th cluster	2427
Bite-5th cluster	2701
Bite-6th cluster	789
Bite-7th cluster	930
Bite-8th cluster	1339

Table 5.13: The number of gestures in kmeans clustered approach.

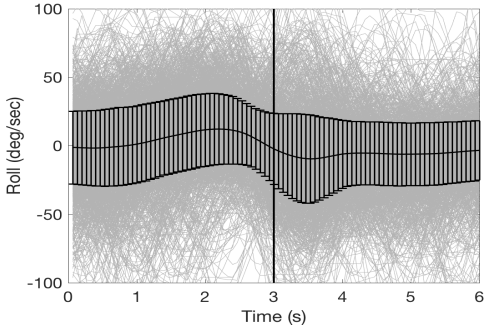
approach. Kmeans clustering partitions N data into k clusters where each data belongs to the nearest cluster [42]. Each vector here is the bite with 6 second duration of roll motion. Figure 5.5 displays the roll motion of eight clusters using kmeans. It can be seen that roll motions from these clusters are quite different. For example, bites from the second, fourth and fifth clusters contain larger motions while bites from the third, sixth and eighth clusters contain relatively small motions. Table 5.13 lists the amount of each gesture.

This yielded 12 total HMMs. During the recognition process, each gesture was passed into 12 HMMs, with gesture type determined by the model with max score. This selection process is similar in Figure 5.2. Notice that if the gesture was recognized as one of the eight bite clusters, it was then determined as "bite".

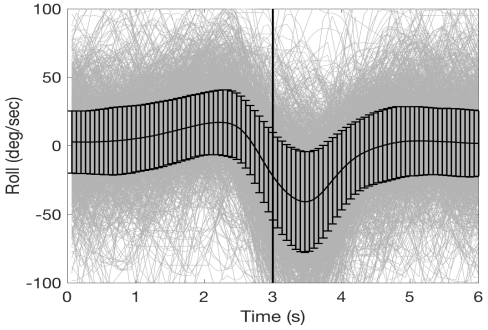


(a) Roll motion of 1st bite cluster.

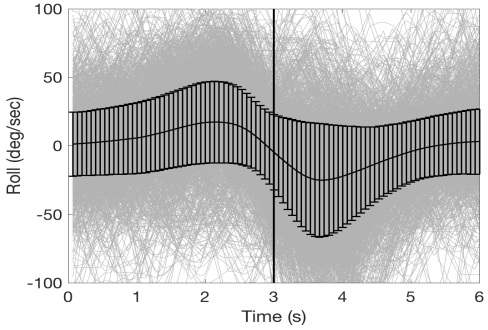
(b) Roll motion of 2nd bite cluster.



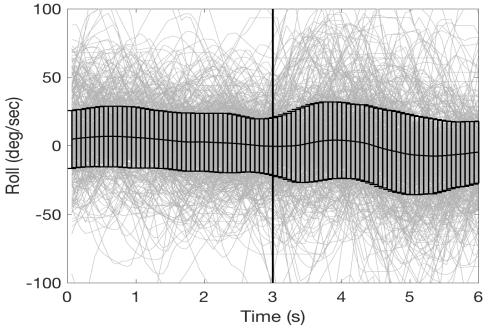
(c) Roll motion of 3rd bite cluster.



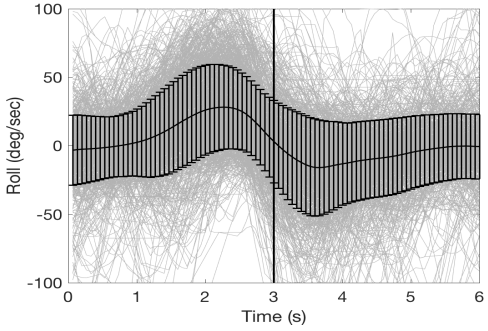
(d) Roll motion of 4th bite cluster.



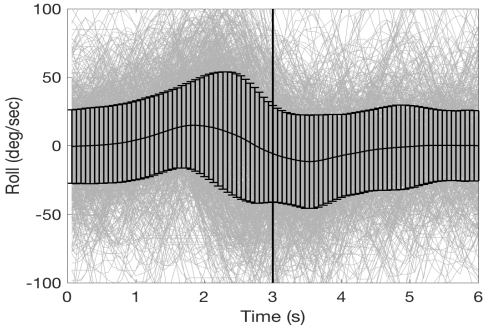
(e) Roll motion of 5th bite cluster.



(f) Roll motion of 6th bite cluster.



(g) Roll motion of 7th bite cluster.



(h) Roll motion of 8th bite cluster.

Figure 5.5: Roll motion of bite from eight clusters. Black curve indicates the averaged roll motion and the standard deviation, gray curves indicate the instances in each cluster.

5.2.2.3 Contextual HMMs with one gesture history.

We investigate the contextual variables along with one gesture history to further improve recognition accuracy. Specifically, a history of one gesture HMM (HMM-1) and bite HMMs (bite variational HMMs, log-score clustered HMM and kmeans clustered HMMs) are combined together. For example, since 9 gestures are determined in the bite variational HMMs, when combined with one history HMM, there are 9 states in total, where each state represents one single gesture. As a result, the prior probability is 9×1 and state transition matrix is 9×9, which are computed as in Equation 4.11 and 4.12. The combinational HMMs of one history with log-score clustered HMM and kmeans clustered HMMs follow the similar protocol.

5.3 Results

Table 5.14 presents the accuracy of the different contextual HMMs. For the baseline classifiers, the accuracy of intake gestures in HMM-S is 85.0% and is improved by 10.0% over the non-HMM algorithm. This demonstrates the effectiveness of temporal dependency of eating gestures. For contextual HMMs, the highest accuracy of all gestures and intake gestures are 86.4% and 91.7%, respectively, with 6.7% and 1.2% improvement over HMM-S. It can also be seen that in the top-down and bottom-up approaches, bite variational HMMs achieve the highest intake gestures accuracy while contain the minimum vocabulary size. This indicates that bite gestures using the same utensil share more similar patterns across the data set. The demographics and meal level HMMs show comparable all gestures accuracy while improving intake gestures accuracy over the HMM-S, ranging between 0.4% and 2.0%.

Table 5.15 presents the accuracy of contextual HMMs with one gesture history. The baseline classifier is the one history HMM (HMM-1) developed in Chapter 4, where each state represents one gesture. The highest intake gestures accuracy of contextual HMMs with one history is 93.0%, with 1.5% improvement over HMM-1. However, the all gestures accuracy is decreased by 0.6%. Most of the decreases exist in rest and utensiling and are failed to recognize as a bite. This can be explained that in the contextual HMMs with one gesture history, there are 12 states in total and 8 states characterize bite. Some of these states represent bites with vigorous motions and are easily confused with utensiling, while some represent bites with small amount of motions and are confused with rest. Although the recognition accuracy of bite is improved, the accuracy of rest and utensiling

decreases. Table 5.16 illustrates the recognition accuracy of each gesture by the different classifiers. The maximum improvements of rest, utensiling, bite, drink and other are: 1.4%, 5.2%, 9.7%, 0.6% and 9.3%, respectively.

It is shown that the all gestures accuracy in contextual HMMs do not improve much. This can be explained by two reasons. One reason is that the recognition accuracy of other ranges only between 40.4% and 61.9%, which drops down the general accuracy. Because of the definition, any motions that do not belong to rest, utensiling, bite and drink are all considered as other, which results in a large motion varieties. For example, other includes gesturing while talking, cleaning with napkin, reaching food, switching plates or containers, adjusting body while eating, etc. Another reason is that some gestures can be confusing. We observe that in many cases a bite involves more head-towards-plate motions when a food is prone to spillage, so a participant positions their head over the container to facilitate delivery of the food to the mouth. These bites are easily confused with rest, because of the small amount of wrist movements caused by head-towards-plate motion. We also observe that it is challenging to recognize gestures with short duration. For example, some participants tend to take quick bites when consume French fries, and the short duration results in inadequate information for the model to recognize that gesture. Figure 5.6 illustrates the number of gestures that are failed to recognize with respect to the amount of contextual HMMs. As the amount of contextual HMMs grows, the number of gestures that are failed to recognize decreases and then reaches the plateau when 7 contextual HMMs are applied, with around 7,000 gestures failed to recognize, which shows the accuracy bound that contextual HMMs could achieve.

5.4 Conclusion

This chapter considers the problem of using contextual variables to improve gesture recognition. We developed hidden Markov models to capture variations in motion patterns using top-down and bottom-up approaches. Specifically, we examined if foreknowledge of the demographics (gender, age, ethnicity, BMI, handedness), meal-level variables (utensil being used, types of foods being eaten), language variables (bite sub-vocabulary, utensil sub-vocabulary and other sub-vocabulary) and clustered based approaches could improve recognition accuracy. Improvement in accuracy was measured by comparing to a non-HMM algorithm and HMM-S that were trained on all participants. The highest accuracy of all gestures and intake gestures in contextual HMMs is 86.4% and 91.7%,

Classifier	Vocabulary size (#HMMs)	All gestures (%)	Intake gestures (%)
Baseline			
Non-HMM baseline	1	-	75.0
HMM-S	5	85.2	85.0
Top-down approach			
Demographics			
Gender HMMs	10	85.1	86.8
Age HMMs	20	84.9	86.1
Handedness HMMs	10	86.0	86.0
Ethnicity HMMs	25	84.6	85.4
BMI HMMs	15	84.4	85.8
Meal level			
Utensil HMMs	25	85.6	87.0
Food HMMs	20	85.0	86.0
Language variations			
Bite variational HMMs	9	86.1	**91.7**
Utensiling variational HMMs	10	85.5	83.6
Other variational HMMs	6	82.6	83.1
Bottom-up approach			
Log-score clustered HMMs	12	86.3	91.3
Kmeans clustered HMMs	12	**86.4**	91.2

Table 5.14: Recognition accuracy of contextual HMMs. The highest accuracy is highlighted.

Classifier	Vocabulary size (#HMMs)	All gestures (%)	Intake gestures (%)
Baseline (HMM-1)	5	**89.5**	91.5
Bite variational HMM-1	9	88.9	**93.0**
Log-score clustered HMM-1	12	86.3	92.0
Kmeans clustered HMM-1	12	88.9	92.0

Table 5.15: Recognition accuracy of contextual HMMs with one gesture history. The highest accuracy is highlighted.

Classifier (HMMs)	Rest (%)	Utensiling (%)	Bite (%)	Drink (%)	Other (%)
HMM-S	86.8	86.9	83.7	96.1	52.6
Gender	84.5	86.1	85.7	96.3	51.3
Age	85.1	86.8	85.2	93.8	43.2
Hand	**88.2**	86.7	84.9	96.2	54.2
Ethnicity	84.6	87.5	84.7	91.5	40.4
BMI	83.9	86.0	84.9	93.8	51.8
Utensil	84.8	87.9	86.2	93.1	48.7
Food	86.5	85.7	85.2	92.9	46.1
Bite variational	86.0	82.2	91.2	95.5	47.4
Utensiling variational	85.3	**92.1**	82.1	96.1	46.2
Other variational	83.3	83.1	81.5	**96.7**	**61.9**
Log-score clustered	84.4	83.5	91.2	94.9	46.2
Kmeans clustered	86.2	83.3	90.9	95.0	51.1
Bite variational HMM-1	85.5	90.0	**93.4**	90.1	53.0
Log-score clustered HMM-1	79.5	88.0	92.5	87.9	55.0
Kmeans clustered HMM-1	87.4	89.6	92.3	89.8	51.9

Table 5.16: Recognition accuracy for five gestures. The highest accuracy of each gesture is highlighted.

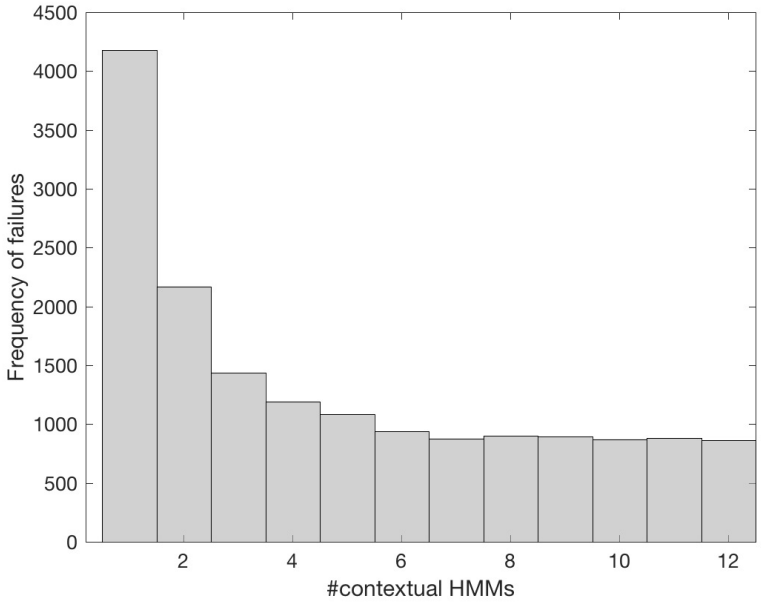


Figure 5.6: Gestures that are failed to recognize with respect to the amount of contextual HMMs.

improved by 1.2% and 6.7%, over HMM-S. In the contextual HMMs with one gesture history, the highest intake gestures accuracy is 93.0% and improved by 1.5% while the all gestures accuracy is 88.9% and decreased by 0.6%, when compared with HMM-S with one gesture history. We also investigate the relation between the number of gestures that are failed to recognize and the amount of contextual HMMs. Result shows that around 7,000 gestures that are consistently failed to recognize, which illustrates the accuracy bound that contextual HMMs could achieve.

Chapter 6

Conclusions

This dissertation is motivated by recent advances in body-worn sensors for automatic monitoring of energy intake [35, 63, 102]. Our group has been investigating using a wrist-worn configuration of sensors to detect periods of eating [27] and track hand-to-mouth gestures [26, 97]. One benefit of wrist-mounted sensors is that they can be embodied in a device that resembles a common watch. This makes the monitoring inconspicuous which helps promote long-term daily use [107].

Wrist-worn wearable devices containing accelerometers and gyroscopes can be used to recognize eating related gestures [10, 26, 52, 107]. Gesture recognition has been widely studied in the domain of sign language recognition [30, 130], motivating a similar approach for eating gesture recognition [97]. However, the variability in motion of an eating gesture is much larger than the variability in motion of a sign language gesture. Eating gestures are a result of a physiological activity (eating) and their execution varies depending on many variables including the subject, utensil, and food or beverage consumed. This dissertation considers the problem of recognizing eating gesture by tracking wrist motion. Eating gestures can have large variability in motion depending on the subject, utensil, and type of food or beverage being consumed. However, it is challenging to build a generic HMM to capture all the variability. This dissertation attempts to build different HMMs of many variables, including demographics (gender, age, ethnicity), utensil being used, or types of foods being eaten, to reduce variations in the wrist motions while eating.

In Chapter 2, a baseline model of a non-HMM method was described to only detects one type of gesture (called bites but includes any food or liquid intake). The method was tested on 276 people eating a meal in a cafeteria and was evaluated on 24,088 bites. It achieved 75% sensitivity and

89% positive predictive value. In Chapter 3, a segment-based method was described to label eating gestures in a large data set. The set of gestures include taking a bite of food (bite), sipping a drink of liquid (drink), manipulating food for preparation of intake (utensiling), and not moving (rest). All other activities such as using a napkin or gesturing while talking are grouped into a non-eating category (other). A total of 18 human raters labeled the same data used described above. Inter-rater reliability was 92.5% demonstrating reasonable consistency of gesture definitions. In Chapter 4, a work was described to explore the complexity of HMMs and the amount of training data needed to adequately capture the motion variability across the large data set introduced in Chapter 3. Results found that HMMs needed a complexity of 13 states and 5 Gaussians to reach a plateau in accuracy, signifying that a minimum of 65 samples per gesture type are needed. Results also found that 500 training samples per gesture type were needed to identify the point of diminishing returns in recognition accuracy. Overall, it achieved 85.2% all gestures accuracy for HMM-S that models a single gesture as a sequence of sub-gestures. It also achieved 89.5% all gestures accuracy for HMM-1, where a sequence of one previous gesture was studied as context. In Chapter 5, a work was described to investigate contextual variables to recognize gestures using top-down and bottom-up approaches. Specifically, we consider if foreknowledge of the demographics (gender, age, hand used, ethnicity, BMI), meal level variables (utensil used for eating, food consumed), language variables (variations of bite, utensiling and other), and clustering based method can improve recognition accuracy. We investigated this hypothesis by building HMMs trained for each of these contextual variables. Results show that the highest accuracy of all gestures and intake gestures in contextual HMMs is 86.4% and 91.7%, improved by 1.2% and 6.7% over HMM-S, respectively. We also investigate the contextual variables along with one gesture history. It achieved all gestures accuracy up to 88.9% and intake gestures accuracy up to 93.0%.

6.1 Future Work

Future work includes automatically classifying and segmenting eating gestures from a continuous wrist motion signal in real-time; estimating the intake calories given the segmented gestures; publishing data set to provide other researchers the opportunity to apply their algorithms; collecting free-living data from individuals over a longer duration (e.g., a week or more) and in multiple locations. This section mainly discusses a tentative method for automatically detecting bite using a

HMM.

6.1.1 Eating and non-eating labels

The first to determine is the start and end time index of each meal, where bites are only detected during that time range. Start index is defined as 10 s prior the first bite while end index is defined as 10 s after the last bite. A sliding window with a step of 1 index is moving and a ± 1 s window centered at each time index is created, where the type of eating activity is determined: eating or non-eating. If the ground truth of gesture type within the window is bite and its duration is more than 70%, it is determined as eating; otherwise it is determined as non-eating. Observables in the ± 1 s window are computed using the same method in Chapter 4.2.3.1.

6.1.2 Training and Testing

Two HMMs are built for eating and non-eating, each with 13 states and 5 Gaussians. We use an architecture of left-to-right with skip and forward-backward algorithm is used to train the models. During testing, observables of each ± 1 s window are passed into HMMs. Each HMM computes the likelihood of this particular observation sequence, indicating how well a model matches the observables. Each unknown activity obtains probability scores from each of the 2 HMMs. Therefore, it is expected that the ratio of eating to non-eating will be high for cases where eating happens within the window, otherwise the ratio will be low. Figure 6.1 shows an example for the ratio of the log score of eating to the log score of non-eating, where a high score indicates high probability of eating activity. It is evident that the peak of ratio appears when the ground truth of "bite" takes place. This demonstrates the potential ability of HMM for eating detection in real-time.

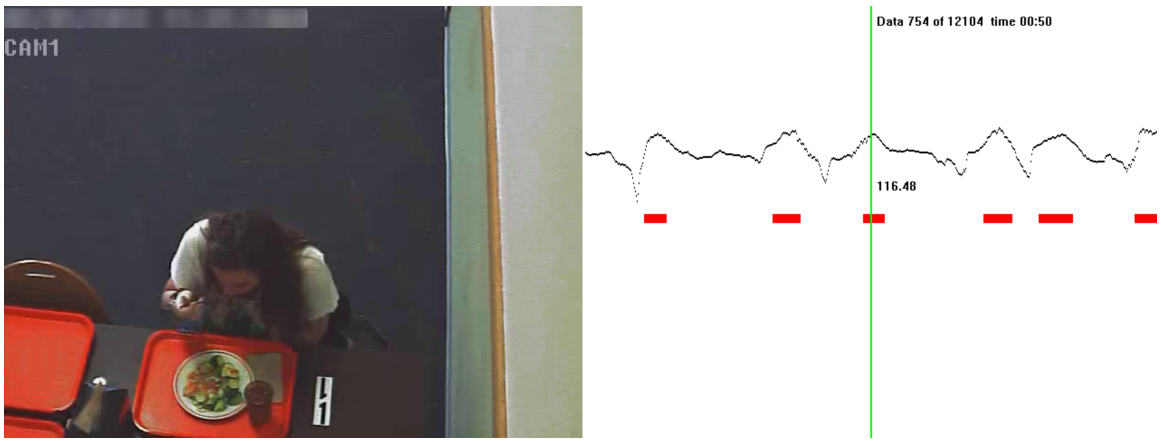


Figure 6.1: Ratio of log score of bite to log score of non-bite. High score indicates high probability of eating activity. Red box indicates groud truth of gesture bite.

Appendices

Appendix A Instructions of using HMM toolbox

We use the MATLAB toolbox developed by Kervin Murphy and his group to build HMM [81]. In this appendix, we introduce the instructions to build a continuous HMM.

A.1 Notation

Q indicates the number of states, O indicates the dimension of observables, M indicates the number of Gaussians, T indicates time duration and nex indicates the number of input sequences. Prior is the prior probability of a HMM, transmat is the state transition matrix. Since continuous HMM is considered here, the emission probability is represented by the μ, Σ and $mixmat$ of the observables.

The prior, transmat,μ, Σ and $mixmat$ are illustrated in Equation 1.

$$\begin{aligned} prior(i) &= P(Q(1) = i) \\ transmat(i,j) &= P(Q(t+1) = j | Q(t) = i) \\ \mu(:,j,k) &= E[Y(t)|Q(t) = j, M(t) = k] \\ Sigma(:,:,j,k) &= Cov[Y(t)|Q(t) = j, M(t) = k] \\ mixmat(j,k) &= P(M(t) = k|Q(t) = j) \end{aligned} \tag{1}$$

If there is only 1 Gaussian, then there is no $mixmat$ matrix.

A.2 Data preparation

First, each input sequence should be the observables with $O \times T$. Second, to make a batch of data containing multiple input sequences, a cell data structure in MATLAB can be use, with the size nex, where the length of each input sequence is variable, and notated as *data*. Besides, a matrix format of data notated as ***data_mtr*** is also prepared for later use, with dimension $O \times T_{tot}$, T_{tot} is the total length of the batched input observables.

A.3 Initialization

In this toolkit, EM algorithm is used to optimize the paramaters in HMM, which only finds a local optimal, thus it is important to choose a good initialization. What we did is to initialize the

μ and Σ of the observables by the estimation from K-means clustering.

```
cov_type = 'diag';
init_method = 'kmeans';
[mu0, Sigma0] = mixgauss_init(Q*M, data_mtr, cov_type, init_method);
mu0 = reshape(mu0, [O Q M]);
Sigma0 = reshape(Sigma0, [O O Q M]);
mixmat0 = mk_stochastic(rand(Q,M));
```

For the simplicity, covariance matrix is constrained to be diagonal, as illustrated as "`cov_type`" in the snippet above.

A.4 HMM training

EM is used to optimize the parameters of HMM:

```
[LL, prior1, transmat1, mu1, Sigma1, mixmat1] = ...
mhmm_em(data, prior0, transmat0, mu0, Sigma0, mixmat0, 'max_iter', 2);
```

where prior0, transmat0, mu0, Sigma0, mixmat0 are the initialized value and prior1, transmat1, mu1, Sigma1, mixmat1 are the values after training, '`max_iter`' is the number of iterations EM run on the training data and LL is the score indicating how well the input sequences match the trained models.

A.5 Evaluating observable sequence on trained models

In our work, multiple HMMs were built for gestures within different contextual groups. During testing, each observable sequence is passed into these HMMs and the log scores are calculated indicating how well the input matches the model. Gesture type of the input is determined by the model with max log score. For example, 5 HMMs were built for rest, utensiling, bite, drink and other, then five log scores are calculated as:

```
rest = mhmm_logprob(data, rest_prior, rest_transmat, rest_mu, rest_sigma, rest_mixmat);
uten = mhmm_logprob(data, uten_prior, uten_transmat, uten_mu, uten_sigma, uten_mixmat);
bite = mhmm_logprob(data, bite_prior, bite_transmat, bite_mu, bite_sigma, bite_mixmat);
drink = mhmm_logprob(data, drink_prior, drink_transmat, drink_mu, drink_sigma, drink_mixmat);
```

```
other = mhmm_logprob(data, other_prior, other_transmat, other_mu, other_sigma, other_mixmat);
```

where prior, transmat, mu, sigma and mixmat are the trained parameters in each HMM. Model with the max score determines gesture type of the test sequence.

A.6 Computing the most probable sequence

Viterbi algorithm is used to find the optimal states of the observable sequence, given the model.

```
obslik = mixgauss_prob(data, mu, Sigma, mixmat);
[pred_state] = viterbi_path(prior, transmat, obslik);
```

where mu, Sigma and mixmat are the trained parameters.

Bibliography

[1] English grammar — Wikipedia, the free encyclopedia. Accessed April 16, 2018.

[2] Hidden markov model — Wikipedia, the free encyclopedia. Accessed April 17, 2018.

[3] Adibi, S. (2012). Link technologies and blackberry mobile health (mhealth) solutions: a review. *IEEE Transactions on Information Technology in Biomedicine*, 16(4):586–597.

[4] Al-Rousan, M., Assaleh, K., and Talaa, A. (2009). Video-based signer-independent arabic sign language recognition using hidden Markov models. *Applied Soft Computing*, 9(3):990–999.

[5] Almaghrabi, R., Villalobos, G., Pouladzadeh, P., and Shirmohammadi, S. (2012). A novel method for measuring nutrition intake based on food image. In *Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International*, pages 366–370. IEEE.

[6] Amft, O. (2010). A wearable earpad sensor for chewing monitoring. In *Sensors, 2010 IEEE*, pages 222–227. IEEE.

[7] Amft, O., Junker, H., and Troster, G. (2005a). Detection of eating and drinking arm gestures using inertial body-worn sensors. In *2005 Proceedings Ninth IEEE International Symposium on Wearable Computers*, pages 160–163.

[8] Amft, O., Stäger, M., Lukowicz, P., and Tröster, G. (2005b). Analysis of chewing sounds for dietary monitoring. In *International Conference on Ubiquitous Computing*, pages 56–72. Springer.

[9] Amft, O. and Troster, G. (2006). Methods for detection and classification of normal swallowing from muscle activation and sound. In *IEEE Pervasive Health Conference and Workshops, 2006*, pages 1–10.

[10] Amft, O. and Troster, G. (2009). On-body sensing solutions for automatic dietary monitoring. *IEEE Pervasive Computing*, 8(2).

[11] Atal, B. and Rabiner, L. (1976). A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 24(3):201–212.

[12] Bai, Y., Jia, W., Mao, Z.-H., and Sun, M. (2014). Automatic eating detection using a proximity sensor. In *Bioengineering Conference (NEBEC), 2014 40th Annual Northeast*, pages 1–2. IEEE.

[13] Barton, A. J. (2012). The regulation of mobile health applications. *BMC medicine*, 10(1):46.

[14] Beattie, V., Edmondson, S., Miller, D., Patel, Y., and Talvola, G. (1995). An integrated multi-dialect speech recognition system with optional speaker adaptation. In *Fourth European Conference on Speech Communication and Technology*.

[15] Bedri, A., Li, R., Haynes, M., Kosaraju, R. P., Grover, I., Prioleau, T., Beh, M. Y., Goel, M., Starner, T., and Abowd, G. (2017). Earbit: Using wearable sensors to detect eating episodes in unconstrained environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):37.

[16] Bi, S., Wang, T., Davenport, E., Peterson, R., Halter, R., Sorber, J., and Kotz, D. (2017). Toward a wearable sensor for eating detection. In *Proceedings of the 2017 Workshop on Wearable Systems and Applications*, pages 17–22. ACM.

[17] Biadsy, F., Hirschberg, J., and Ellis, D. P. (2011). Dialect and accent recognition using phonetic-segmentation supervectors. In *Twelfth Annual Conference of the International Speech Communication Association*.

[18] Chen, L., Hoey, J., Nugent, C. D., Cook, D. J., and Yu, Z. (2012). Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):790–808.

[19] Childers, D. G., Hahn, M., and Larar, J. (1989). Silent and voiced/unvoiced/mixed excitation (four-way) classification of speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(11):1771–1774.

[20] Choudhury, T., Consolvo, S., Harrison, B., Hightower, J., LaMarca, A., LeGrand, L., Rahimi, A., Rea, A., Bordello, G., Hemingway, B., et al. (2008). The mobile sensing platform: An embedded activity recognition system. *IEEE Pervasive Computing*, 7(2).

[21] Chun, K. S., Bhattacharya, S., and Thomaz, E. (2018). Detecting eating episodes by tracking jawbone movements with a non-contact wearable sensor. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(1):4.

[22] Chung, J., Chung, J., Oh, W., Yoo, Y., Lee, W. G., and Bang, H. (2017). A glasses-type wearable device for monitoring the patterns of food intake and facial activity. *Scientific Reports*, 7:41690.

[23] De Castro, J. M. (2000). Eating behavior: lessons from the real world of humans. *Nutrition*, 16(10):800–813.

[24] Dodds, W. J. (1989). The physiology of swallowing. *Dysphagia*, 3(4):171–178.

[25] Dong, Y., Hoover, A., and Muth, E. (2009). A device for detecting and counting bites of food taken by a person during eating. In *Bioinformatics and Biomedicine, 2009. BIBM'09. IEEE International Conference on*, pages 265–268. IEEE.

[26] Dong, Y., Hoover, A., Scisco, J., and Muth, E. (2012). A new method for measuring meal intake in humans via automated wrist motion tracking. *Applied Psychophysiology and Biofeedback*, 37(3):205–215.

[27] Dong, Y., Scisco, J., Wilson, M., Muth, E., and Hoover, A. (2014). Detecting periods of eating during free-living by tracking wrist motion. *IEEE Journal of Biomedical and Health Informatics*, 18(4):1253–1260.

[28] Doulah, A. and Sazonov, E. (2017). Clustering of food intake images into food and non-food categories. In *International Conference on Bioinformatics and Biomedical Engineering*, pages 454–463. Springer.

[29] Eng, D. S. and Lee, J. M. (2013). The promise and peril of mobile health applications for diabetes and endocrinology. *Pediatric diabetes*, 14(4):231–238.

[30] Fang, G., Gao, W., and Zhao, D. (2007). Large-vocabulary continuous sign language recognition based on transition-movement models. *IEEE Transactions on Systems, Man, and Cybernetics-part A: Systems and Humans*, 37(1):1–9.

[31] Farooq, M. and Sazonov, E. (2016). A novel wearable device for food intake and physical activity recognition. *Sensors*, 16(7):1067.

[32] Feldman, R. S. and Rimé, B. (1991). *Fundamentals of nonverbal behavior*. Cambridge University Press.

[33] Finkelstein, E. A., Khavjou, O. A., Thompson, H., Trogdon, J. G., Pan, L., Sherry, B., and Dietz, W. (2012). Obesity and severe obesity forecasts through 2030. *American journal of preventive medicine*, 42(6):563–570.

[34] Finkelstein, E. A., Trogdon, J. G., Cohen, J. W., and Dietz, W. (2009). Annual medical spending attributable to obesity: payer-and service-specific estimates. *Health affairs*, 28(5):w822–w831.

[35] Fontana, J. M. and Sazonov, E. (2015). Detection and characterization of food intake by wearable sensors. In *Wearable Sensors*, pages 591–616.

[36] G. Xuan, W. Zhang, and P. Chai (2001). Em algorithms of gaussian mixture model and hidden markov model. In *International Conference on Imgage Processing*, pages 145–148, Thessaloniki.

[37] Gao, Y., Zhang, N., Wang, H., Ding, X., Ye, X., Chen, G., and Cao, Y. (2016). ihear food: Eating detection using commodity bluetooth headsets. In *Connected Health: Applications, Systems and Engineering Technologies (CHASE), 2016 IEEE First International Conference on*, pages 163–172. IEEE.

[38] Gemming, L., Doherty, A., Utter, J., Shields, E., and Mhurchu, C. N. (2015). The use of a wearable camera to capture and categorise the environmental and social context of self-identified eating episodes. *Appetite*, 92:118–125.

[39] Gibbon, D., Moore, R., and Winski, R. (1997). *Handbook of standards and resources for spoken language systems*. Walter de Gruyter.

[40] Harrell, F. E., Lee, K. L., and Mark, D. B. (1996). Multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine*, 15(4):361–387.

[41] Harrell Jr, F. E., Lee, K. L., Califf, R. M., Pryor, D. B., and Rosati, R. A. (1984). Regression modelling strategies for improved prognostic prediction. *Statistics in Medicine*, 3(2):143–152.

[42] Hartigan, J. A. and Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108.

[43] Hedley, A. A., Ogden, C. L., Johnson, C. L., Carroll, M. D., Curtin, L. R., and Flegal, K. M. (2004). Prevalence of overweight and obesity among us children, adolescents, and adults, 1999-2002. *Jama*, 291(23):2847–2850.

[44] Hins, J., Series, F., Almeras, N., and Tremblay, A. (2006). Relationship between severity of nocturnal desaturation and adaptive thermogenesis: preliminary data of apneic patients tested in a whole-body indirect calorimetry chamber. *International journal of obesity*, 30(3):574.

[45] Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., and Wood, K. (2006). Sensecam: A retrospective memory aid. In *International Conference on Ubiquitous Computing*, pages 177–193. Springer.

[46] Hoover, A., Muth, E., and Dong, Y. (2012). Weight control device using bites detection. US Patent 8,310,368.

[47] Huang, C., Chang, E., Zhou, J., and Lee, K.-F. (2000). Accent modeling based on pronunciation dictionary adaptation for large vocabulary mandarin speech recognition. In *Sixth International Conference on Spoken Language Processing*.

[48] Huang, C., Chen, T., and Chang, E. (2004). Accent issues in large vocabulary continuous speech recognition. *International Journal of Speech Technology*, 7(2-3):141–153.

[49] Huang, Z. (2013). An assessment of the accuracy of an automated bite counting method in a cafeteria setting. Master's thesis, Clemson University, Clemson.

[50] Jia, W., Chen, H.-C., Yue, Y., Li, Z., Fernstrom, J., Bai, Y., Li, C., and Sun, M. (2014). Accuracy of food portion size estimation from digital pictures acquired by a chest-worn camera. *Public Health Nutrition*, 17(8):1671–1681.

[51] Joo, N.-S. and Kim, B.-T. (2007). Mobile phone short message service messaging for behaviour modification in a community-based weight control programme in korea. *Journal of Telemedicine and Telecare*, 13(8):416–420.

[52] Junker, H., Amft, O., Lukowicz, P., and Tröster, G. (2008). Gesture spotting with body-worn inertial sensors to detect user activities. *Pattern Recognition*, 41(6):2010–2024.

[53] Kalantarian, H. and Sarrafzadeh, M. (2015). Audio-based detection and evaluation of eating behavior using the smartwatch platform. *Computers in Biology and Medicine*, 65:1–9.

[54] Kang, H., Lee, C. W., and Jung, K. (2004). Recognition-based gesture spotting in video games. *Pattern Recognition Letters*, 25(15):1701–1714.

[55] Kat, L. W. and Fung, P. (1999). Fast accent identification and accented speech recognition. In *Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on*, volume 1, pages 221–224. IEEE.

[56] Kim, H.-J., Kim, M., Lee, S.-J., and Choi, Y. S. (2012). An analysis of eating activities for automatic food type recognition. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, pages 1–5. IEEE.

[57] Kodama, C., Kato, K., Tamura, S., and Hayamizu, S. (2017). Swallowing function evaluation using deep-learning-based acoustic signal processing. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017*, pages 961–964. IEEE.

[58] Kumar, P., Gauba, H., Roy, P. P., and Dogra, D. P. (2017). Coupled hmm-based multi-sensor data fusion for sign language recognition. *Pattern Recognition Letters*, 86:1–8.

[59] Kumpf, K. and King, R. W. (1996). Automatic accent classification of foreign accented australian english speech. In *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 3, pages 1740–1743. IEEE.

[60] Kwapisz, J. R., Weiss, G. M., and Moore, S. A. (2011). Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82.

[61] L. Rabiner and M. Hill (1989). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286.

[62] Lara, O. D. and Labrador, M. A. (2013). A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys and Tutorials*, 15(3):1192–1209.

[63] Lara, O. D., Labrador, M. A., et al. (2013). A survey on human activity recognition using wearable sensors. *IEEE Communications Surveys and Tutorials*, 15(3):1192–1209.

[64] Lausberg, H. and Sloetjes, H. (2009). Coding gestural behavior with the NEUROGES-ELAN system. *Behavior research methods*, 41(3):841–849.

[65] Lee, C.-H., Soong, F. K., and Paliwal, K. K. (2012). *Automatic speech and speaker recognition: advanced topics*, volume 355. Springer Science & Business Media.

[66] Leonard, R. (1984). A database for speaker-independent digit recognition. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'84.*, volume 9, pages 328–331. IEEE.

[67] Liang, Y. and Li, J. (2017). Computer vision-based food calorie estimation: dataset, method, and experiment. *arXiv preprint arXiv:1705.07632*.

[68] Liu, J., Johns, E., Atallah, L., Pettitt, C., Lo, B., Frost, G., and Yang, G.-Z. (2012). An intelligent food-intake monitoring system using wearable sensors. In *Wearable and Implantable Body Sensor Networks (BSN), 2012 Ninth International Conference on*, pages 154–160. IEEE.

[69] Lopes, I. M., Silva, B. M., Rodrigues, J. J., Lloret, J., and Proença, M. L. (2011). A mobile health monitoring solution for weight control. In *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*, pages 1–5. IEEE.

[70] Lopez-Meyer, P., Makeyev, O., Schuckers, S., Melanson, E. L., Neuman, M. R., and Sazonov, E. (2010). Detection of food intake from swallowing sequences by supervised and unsupervised methods. *Annals of Biomedical Engineering*, 38(8):2766–2774.

[71] Makeyev, O., Lopez-Meyer, P., Schuckers, S., Besio, W., and Sazonov, E. (2012). Automatic food intake detection based on swallowing sounds. *Biomedical Signal Processing and Control*, 7(6):649–656.

[72] Malnick, S. D. and Knobler, H. (2006). The medical complications of obesity. *Journal of the Association of Physicians*, 99(9):565–579.

[73] Mantyjarvi, J., Himberg, J., and Seppanen, T. (2001). Recognizing human motion with multiple acceleration sensors. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, volume 2, pages 747–752. IEEE.

[74] Mari, J.-F., Haton, J.-P., and Kriouile, A. (1997). Automatic word recognition based on second-order hidden markov models. *IEEE Transactions on speech and Audio Processing*, 5(1):22–25.

[75] Martin, C. K., Han, H., Coulon, S. M., Allen, H. R., Champagne, C. M., and Anton, S. D. (2008). A novel method to remotely measure food intake of free-living individuals in real time: the remote food photography method. *British Journal of Nutrition*, 101(3):446–456.

[76] Merck, C., Maher, C., Mirtchouk, M., Zheng, M., Huang, Y., and Kleinberg, S. (2016). Multimodality sensing for eating recognition. In *Proceedings of the 10th EAI International Conference on Pervasive Computing Technologies for Healthcare*, pages 130–137. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[77] Meyers, A., Johnston, N., Rathod, V., Korattikara, A., Gorban, A., Silberman, N., Guadarrama, S., Papandreou, G., Huang, J., and Murphy, K. P. (2015). Im2calories: towards an automated mobile vision food diary. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1233–1241.

[78] Minnen, D., Starner, T., Ward, J. A., Lukowicz, P., and Troster, G. (2005). Recognizing and discovering human actions from on-body sensor data. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1545–1548. IEEE.

[79] Mirtchouk, M., Lustig, D., Smith, A., Ching, I., Zheng, M., and Kleinberg, S. (2017). Recognizing eating from body-worn sensors: Combining free-living and laboratory data. *Proc. of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):85.

[80] Mohandes, M., Deriche, M., Johar, U., and Ilyas, S. (2012). A signer-independent arabic sign language recognition system using face detection, geometric features, and a hidden markov model. *Computers & Electrical Engineering*, 38(2):422–433.

[81] Murphy, K. (1998). Hmm toolbox for matlab. *Internet: http://www. cs. ubc. ca/˜ murphyk/Software/HMM/hmm. html,[Oct. 29, 2011]*.

[82] Nishimura, J. and Kuroda, T. (2008). Eating habits monitoring using wireless wearable in-ear microphone. In *Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on*, pages 130–132. IEEE.

[83] Ogden, C. L., Carroll, M. D., Curtin, L. R., McDowell, M. A., Tabak, C. J., and Flegal, K. M. (2006). Prevalence of overweight and obesity in the united states, 1999-2004. *Jama*, 295(13):1549–1555.

[84] Olubanjo, T. and Ghovanloo, M. (2014). Real-time swallowing detection based on tracheal acoustics. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4384–4388. IEEE.

[85] Papapanagiotou, V., Diou, C., Zhou, L., Van Den Boer, J., Mars, M., and Delopoulos, A. (2017a). A novel chewing detection system based on PPG, audio, and accelerometry. *IEEE Journal of Biomedical and Health Informatics*, 21(3):607–618.

[86] Papapanagiotou, V., Diou, C., Zhou, L., Van Den Boer, J., Mars, M., and Delopoulos, A. (2017b). A novel chewing detection system based on ppg, audio, and accelerometry. *IEEE Journal of Biomedical and Health Informatics*, 21(3):607–618.

[87] Parkka, J., Ermes, M., Korpipaa, P., Mantyjarvi, J., Peltola, J., and Korhonen, I. (2006). Activity classification using realistic data from wearable sensors. *IEEE Transactions on information technology in biomedicine*, 10(1):119–128.

[88] Paßler, S. and Fischer, W.-J. (2014). Food intake monitoring: Automated chew event detection in chewing sounds. *IEEE Journal of Biomedical and Health Informatics*, 18(1):278–289.

[89] Paßler, S., Fischer, W.-J., and Kraljevski, I. (2012). Adaptation of models for food intake sound recognition using maximum a posteriori estimation algorithm. In *Wearable and Implantable Body Sensor Networks (BSN), 2012 Ninth International Conference on*, pages 148–153. IEEE.

[90] Pavlovic, V. I., Sharma, R., and Huang, T. S. (1997). Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695.

[91] Peduzzi, P., Concato, J., Kemper, E., Holford, T. R., and Feinstein, A. R. (1996). A simulation study of the number of events per variable in logistic regression analysis. *Journal of Clinical Epidemiology*, 49(12):1373–1379.

[92] Pettitt, C., Liu, J., Kwasnicki, R. M., Yang, G.-Z., Preston, T., and Frost, G. (2016). A pilot study to determine whether using a lightweight, wearable micro-camera improves dietary assessment accuracy and offers information on macronutrients and eating rate. *British Journal of Nutrition*, 115(1):160–167.

[93] Petty, A. J., Melanson, K. J., and Greene, G. W. (2013). Self-reported eating rate aligns with laboratory measured eating rate but not with free-living meals. *Appetite*, 63:36–41.

[94] Prioleau, T., Moore II, E., and Ghovanloo, M. (2017). Unobtrusive and wearable systems for automatic dietary monitoring. *IEEE Transactions on Biomedical Engineering*, 64(9):2075–2089.

[95] Puri, M., Zhu, Z., Yu, Q., Divakaran, A., and Sawhney, H. (2009). Recognition and volume estimation of food intake using a mobile device. In *Applications of Computer Vision (WACV), 2009 Workshop on*, pages 1–8. IEEE.

[96] Rahman, S. A., Merck, C., Huang, Y., and Kleinberg, S. (2015). Unintrusive eating recognition using google glass. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2015 9th International Conference on*, pages 108–111. IEEE.

[97] Ramos-Garcia, R. I., Muth, E. R., Gowdy, J. N., and Hoover, A. W. (2015). Improving the recognition of eating gestures using intergesture sequential dependencies. *IEEE Journal of Biomedical and Health Informatics*, 19(3):825–831.

[98] Rautaray, S. S. and Agrawal, A. (2015). Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54.

[99] Salley, J. N., Hoover, A. W., Wilson, M. L., and Muth, E. R. (2016). Comparison between human and bite-based methods of estimating caloric intake. *Journal of the Academy of Nutrition and Dietetics*, 116(10):1568–1577.

[100] Sazonov, E., Schuckers, S., Lopez-Meyer, P., Makeyev, O., Sazonova, N., Melanson, E. L., and Neuman, M. (2008). Non-invasive monitoring of chewing and swallowing for objective quantification of ingestive behavior. *Physiological measurement*, 29(5):525.

[101] Sazonov, E. S. and Fontana, J. M. (2012). A sensor system for automatic detection of food intake through non-invasive monitoring of chewing. *IEEE Sensors Journal*, 12(5):1340–1348.

[102] Sazonov, E. S. and Schuckers, S. (2010). The energetics of obesity: A review: Monitoring energy intake and energy expenditure in humans. *IEEE Engineering in Medicine and Biology Magazine*, 29(1):31–35.

[103] Schoeller, D. A. (1995). Limitations in the assessment of dietary energy intake by self-report. *Metabolism-Clinical and Experimental*, 44:18–22.

[104] Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press.

[105] Scisco, J. L., Muth, E. R., and Hoover, A. W. (2014). Examining the utility of a bite-count–based measure of eating activity in free-living human beings. *Journal of the Academy of Nutrition and Dietetics*, 114(3):464–469.

[106] Shen, Y., Muth, E., and Hoover, A. (2016). Recognizing eating gestures using context dependent hidden markov models. In *Connected Health: Applications, Systems and Engineering Technologies (CHASE), 2016 IEEE First International Conference on*, pages 248–253. IEEE.

[107] Shen, Y., Salley, J., Muth, E., and Hoover, A. (2017). Assessing the accuracy of a wrist motion tracking method for counting bites across demographic and food variables. *IEEE Journal of Biomedical and Health Informatics*, 21(3):599–606.

[108] Shoaib, M., Bosch, S., Scholten, H., Havinga, P. J., and Incel, O. D. (2015). Towards detection of bad habits by fusing smartphone and smartwatch sensors. In *Pervasive Computing and Communication Workshops (PerCom Workshops), 2015 IEEE International Conference on*, pages 591–596. IEEE.

[109] Shuzo, M., Komori, S., Takashima, T., Lopez, G., Tatsuta, S., Yanagimoto, S., Warisawa, S., Delaunay, J.-J., and Yamada, I. (2010). Wearable eating habit sensing system using internal body sound. *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, 4(1):158–166.

[110] Singh, M., Mandal, M., and Basu, A. (2005). Visual gesture recognition for ground air traffic control using the radon transform. In *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 2586–2591. IEEE.

[111] Starner, T. and Pentland, A. (1997). Real-time american sign language recognition from video using hidden markov models. In *Motion-Based Recognition*, pages 227–243. Springer.

[112] Starner, T., Weaver, J., and Pentland, A. (1998). Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375.

[113] Subar, A. F., Kirkpatrick, S. I., Mittl, B., Zimmerman, T. P., Thompson, F. E., Bingley, C., Willis, G., Islam, N. G., Baranowski, T., McNutt, S., et al. (2012). The automated self-administered 24-hour dietary recall (asa24): a resource for researchers, clinicians, and educators from the national cancer institute. *Journal of the Academy of Nutrition and Dietetics*, 112(8):1134–1137.

[114] Sun, M., Burke, L. E., Mao, Z.-H., Chen, Y., Chen, H.-C., Bai, Y., Li, Y., Li, C., and Jia, W. (2014). ebutton: a wearable computer for health monitoring and personal assistance. In *Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE*, pages 1–6. IEEE.

[115] Suykens, J. A. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300.

[116] Thomas, J. G. and Bond, D. S. (2014). Review of innovations in digital health technology to promote weight control. *Current diabetes reports*, 14(5):485.

[117] Thomaz, E., Bedri, A., Prioleau, T., Essa, I., and Abowd, G. D. (2017). Exploring symmetric and asymmetric bimanual eating detection with inertial sensors on the wrist. In *Proc. of the ACM on the 1st Workshop on Digital Biomarkers*, pages 21–26.

[118] Thomaz, E., Essa, I., and Abowd, G. D. (2015). A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1029–1040. ACM.

[119] Tomokiyo, L. M. and Waibel, A. (2003). Adaptation methods for non-native speech. *Multilingual Speech and Language Processing*, 6.

[120] Torres-Carrasquillo, P. A., Gleason, T. P., and Reynolds, D. A. (2004). Dialect identification using gaussian mixture models. In *ODYSSEY04-The Speaker and Language Recognition Workshop*.

[121] Tsai, C. C., Lee, G., Raab, F., Norman, G. J., Sohn, T., Griswold, W. G., and Patrick, K. (2007). Usability and feasibility of pmeb: a mobile phone application for monitoring real time caloric balance. *Mobile Networks and Applications*, 12(2-3):173–184.

[122] Villalobos, G., Almaghrabi, R., Pouladzadeh, P., and Shirmohammadi, S. (2012). An image procesing approach for calorie intake measurement. In *Medical Measurements and Applications Proceedings (MeMeA), 2012 IEEE International Symposium on*, pages 1–5. IEEE.

[123] Wachs, J. P., Kölsch, M., Stern, H., and Edan, Y. (2011). Vision-based hand-gesture applications. *Communications of the ACM*, 54(2):60–71.

[124] Walker, W. P. and Bhatia, D. (2011). Towards automated ingestion detection: Swallow sounds. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 7075–7078. IEEE.

[125] Wang, Z., Schultz, T., and Waibel, A. (2003). Comparison of acoustic model adaptation techniques on non-native speech. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–I. IEEE.

[126] Wu, X.-H., Su, M.-C., and Wang, P.-C. (2010). A hand-gesture-based control interface for a car-robot. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4644–4648. IEEE.

[127] Yang, H.-D., Park, A.-Y., and Lee, S.-W. (2007). Gesture spotting and recognition for human–robot interaction. *IEEE Transactions on Robotics*, 23(2):256–270.

[128] Yatani, K. and Truong, K. N. (2012). Bodyscope: a wearable acoustic sensor for activity recognition. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 341–350. ACM.

[129] Ye, X., Chen, G., and Cao, Y. (2015). Automatic eating detection using head-mount and wrist-worn accelerometers. In *E-health Networking, Application & Services (HealthCom), 2015 17th International Conference on*, pages 578–581. IEEE.

[130] Zaki, M. M. and Shaheen, S. I. (2011). Sign language recognition using a combination of new vision based features. *Pattern Recognition Letters*, 32(4):572–577.

[131] Zhang, R. and Amft, O. (2016). Bite glasses: Measuring chewing using emg and bone vibration in smart eyeglasses. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, pages 50–52. ACM.

[132] Zhang, R. and Amft, O. (2018). Monitoring chewing and eating in free-living using smart eyeglasses. *IEEE Journal of Biomedical and Health Informatics*, 22(1):23–32.

[133] Zhang, R., Bernhart, S., and Amft, O. (2016a). Diet eyeglasses: Recognising food chewing using emg and smart eyeglasses. In *Wearable and Implantable Body Sensor Networks (BSN), 2016 IEEE 13th International Conference on*, pages 7–12. IEEE.

[134] Zhang, R., Bernhart, S., and Amft, O. (2016b). Diet eyeglasses: Recognising food chewing using EMG and smart eyeglasses. In *Wearable and Implantable Body Sensor Networks (BSN), 2016 IEEE 13th International Conference on*, pages 7–12. IEEE.

[135] Zhang, S., Ang, M. H., Xiao, W., and Tham, C. K. (2009). Detection of activities by wireless sensors for daily life surveillance: eating and drinking. *Sensors*, 9(3):1499–1517.