Experiences in Building a Real-World Eating Recogniser [*]

Sougata Sen, Vigneshwaran Subbaraju, Archan Misra, Rajesh Krishna Balan, Youngki Lee
School of Information Systems, Singapore Management University
{sougata.sen.2012,vigneshwaran,archanm,rajesh,youngkilee}@smu.edu.sg

ABSTRACT

In this paper, we describe the progressive design of the gesture recognition module of an automated food journaling system – *Annapurna*. *Annapurna* runs on a smartwatch and utilises data from the inertial sensors to first identify eating gestures, and then captures food images which are presented to the user in the form of a food journal. We detail the lessons we learnt from multiple in-the-wild studies, and show how eating recognizer is refined to tackle challenges such as (i) high gestural diversity, and (ii) non-eating activities with similar gestural signatures. *Annapurna* is finally robust (identifying eating across a wide diversity in food content, eating styles and environments) and accurate (false-positive and false-negative rates of 6.5% and 3.3% respectively).

1. INTRODUCTION

Gesture recognition, based on the inertial sensors embedded in wearable devices, has gained increasing popularity recently. Such gesture recognition techniques have been used to identify gesture-driven lifestyle activities such as smoking [7] and eating [1]. In particular, unobtrusive wearable-based solutions [11, 12] for eating detection are of strong interest, as they can help in losing or maintaining target weight, or capturing irregular habits such as eating too fast or skipping meals.

Broadly speaking, research in the area of automated eating gesture detection and diet monitoring has two goals: (a) Identifying the eating gesture (e.g. [1, 2, 12]), or (b) Identifying the food item consumed (e.g. [3, 9]). We have recently focused on building an end-to-end wearable-based system (called *Annapurna*), which aims to unobtrusively build a food journal by automatically capturing images of the food items consumed by a user throughout the day. Motivated by the popularity of smartwatches (with models such as Samsung Gear 1 & Gear 2 containing an embedded camera), our core idea is (a) to use the inertial sensors on the smartwatch to identify the eating-related "hand-to-mouth" gestures; and (b) to additionally use the embedded smartwatch camera to capture images of the food being consumed.

In this paper, we focus on *Annapurna's* first challenge: building a robust eating gesture recognition module, which can identify real-world eating gestures. We describe our *experiences* with the iterative design and deployment of *Annapurna* to real users. Through both extensive controlled studies (21 users, 5 nationalities, 135 eating episodes) and multiple in-the-wild deployments (7 users, 12 days, total of 78 meals), we discover the following key challenges and principles (which are likely to apply to a broader class of continuous gesture-driven lifestyle activity monitoring services):

- *Diversity of Eating Gestures:* We find and demonstrate that eating is very diverse activity, with differences related to: (a) food type (e.g., rice vs. noodles, sandwiches, burgers, soups, etc.), (b) environment (e.g., type of seating, height of table , etc.), (c) eating styles and mode of eating (e.g., with chopsticks, forks, using hands). The resulting differences in the trajectory of corresponding hand-to-mouth gestures makes it difficult to build a high-accuracy eating gesture recognizer.

- *Confusion with similar Real-world Lifestyle Activities:* Through our studies, we find that real-world users perform a variety of other non-eating activities (e.g., smoking, drinking, washing one's face or putting on makeup) which give rises to gestures that are similar to eating. These gestures led to a false positive rate that was much higher than that encountered in our realistic, but controlled studies, and necessitated *significant* enhancements to the base classification model.

- *Inability to Track Singleton Gestures:* We find that, in real world, it is impossible to identify each and every eating gesture. Consequently, we focused on detecting meal episodes, which consist of *multiple*, repeated eating gestures. Moreover, we had to abandon our initial goal of gesture instance-triggered image capture, and instead used a more bursty image capture approach that clicks images continuously, once a meal episode is detected.

Overall, our experiences show that it is still possible to

[*] This work was supported partially by Singapore Ministry of Education Academic Research Fund Tier 2 under research grant MOE2011-T2-1001, and partially by the National Research Foundation, Prime Minister's Office, Singapore, under IDM Futures Funding Initiative. All findings and recommendations are those of the authors and do not necessarily reflect the views of the granting agency, or Singapore Management University.

WPA'17, June 19, 2017, Niagara Falls, NY, USA
© 2017 ACM. ISBN 978-1-4503-4958-1/17/06. . . $15.00
DOI: http://dx.doi.org/10.1145/3092305.3092306

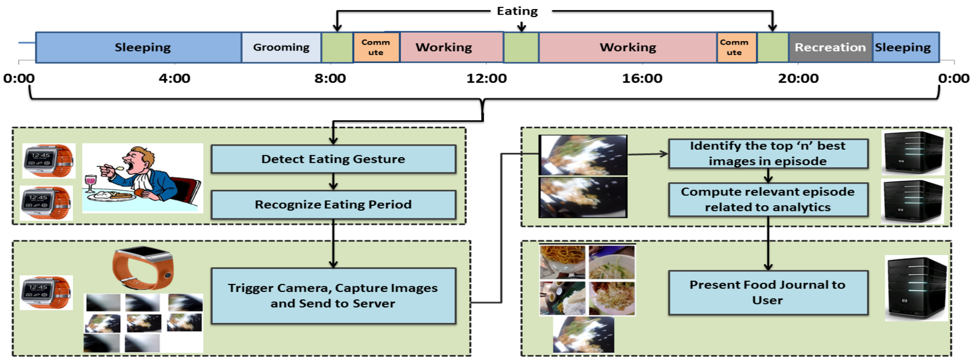

Eating
Sleeping
Grooming
Comm ute
Working
Working
Comm ute
Recreation
Sleeping
0:00
4:00
8:00
12:00
16:00
20:00
0:00
Detect Eating Gesture
Recognize Eating Period
Trigger Camera, Capture Images and Send to Server
Identify the top 'n' best images in episode
Compute relevant episode related to analytics
Present Food Journal to User

Figure 1: Overview of *Annapurna*

build a robust gesture recognizer to detect *meals*. In particular, we shall describe *Annapurna's* implementation of a low-energy, robust and real-time classifier (on a resource-limited smartwatch) using a 2-tier classifier that: (a) uses a low-pass accelerometer stream (2.5 sec frames) to identify potential eating episodes; and (b) confirms this possibility by detecting multiple successive eating gestures. This system has an overall precision of 93.5% for detecting eating episodes (meals) in the real-world, and a low false negative rate of 3.3%.

2. INITIAL ANALYSIS & DESIGN GOALS

While our prior work [11] demonstrated the feasibility of eating detection using a wrist-worn device gestures, a fully deployable automated food journaling system must address additional other requirements. In this section, we describe the *Annapurna's* design goals, focusing especially on eating detection component, along with pertinent insights gained from observing multiple real-world eating episodes.

2.1 System Overview

Annapurna is an automated food journaling system, that provides a user wearing a wrist-worn smartwatch with curated images of the food that she has consumed during at various meals during the day. Figure 1 shows an overview of the system, which consists of the following components:

Eating Gesture Recognizer: The eating gesture recognition component continuously runs on the smartwatch, utilising the inertial sensor data to determine the hand-to-mouth gestures as well as eating periods. The eating gesture recogniser should accommodate the variations in the sensors readings introduced by the diversity of users and eating styles (e.g. see Figure 2). It must both have low false negatives (not miss any of the eating episodes) and low false positives (i.e. not mistakenly classify other similar gestural activities as 'eating').

Image Capturing & Processing: Once the onset of an eating episode has been identified (multiple closely spaced eating gestures), the smartwatch captures images automatically and unobtrusively. Once the images are captured, they are sent to a backend server, where various image processing techniques are applied to eliminate irrelevant images. The entire process is optimised for energy consumption.

Food Journaling: Finally, a small subset of relevant images corresponding to an eating episode is stored in the server. The user can view these these images, as well as other eating related statistics, via a Web portal.

2.2 Design Goals

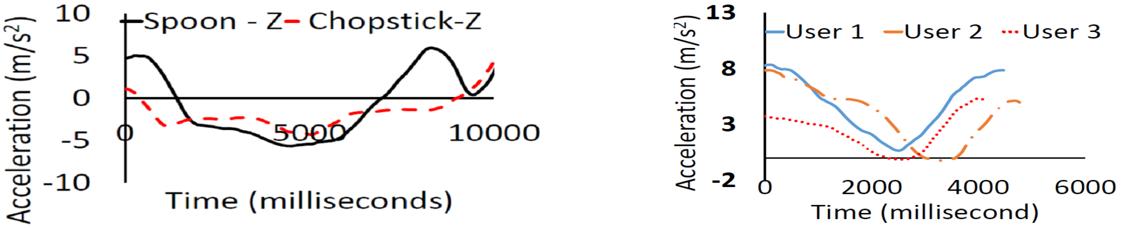
Annapurna's gesture recognizer is designed to accommodate the following characteristics:

Focus only on persistent eating episodes that last at least 5 minutes: Most eating episodes are not fleeting (they last several minutes) and consist of multiple hand-to-mouth gestures. Hence, our eating detector need not detect each individual eating gesture, but can utilize longer observation windows for robustness. We do not try to track extremely transient eating activities (e.g., consuming a single candy).

Tolerate diversity in eating style, and gesture duration: Our studies show the existence of considerable gestural diversity in eating-based on not just individual level behavior, but also the content of the food. Figure 2a shows the accelerometer trace for 2 different eating modes (spoon and chopstick), whereas Figure 2b shows the traces for 3 different individuals eating rice. We see that the gesture traces are quite different, both across eating modes and food types. Our design must accommodate such diversity.

Focus only on plate-related eating episodes: Because the gesture detection is followed by a process of capture of food images, we can focus on detecting eating episodes that involve some utensil. More specifically, we do not explicitly target scenarios where the food is consumed on-the-go–e.g., a user is walking and eating a sandwich, as the smartwatch camera is unlikely to obtain an image of such food items.

The gesture recognizer makes the following additional *assumptions*: (1) We assume that the user wears the smartwatch on the dominant hand while eating. While watches are often worn on the non-dominant hand, wearable such as fitness bands are gaining in popularity. Hence, this might not be a major limitation. Moreover, for certain eating styles (e.g. knife in the dominant hand and fork on the non-dominant hand), this might not be a limitation. (2) Since the overall goal of the *Annapurna* system is to capture images of the food plate, we assume that the food is served on plates or containers. However, we do not limit the type of food consumed – e.g., we can capture fast-food items, as long as the user interacts with a container containing that food. It must be noted that even though this assumption does not affect the eating gesture recognition system (eating gesture identification does not rely on the container), it affects the overall system goal of capturing images of the food plate.



(a) Accelerometer's Z-axis variation during eating with spoon v chopstick.

(b) Accelerometer's Z-axis depicting difference in gesture across individuals.

Figure 2: Diversity across Eating Styles & Users

User Study	(Users,) Duration	Eating Detector	TP	FP (only inertial)	FN
1	7 users, 5 days	Light-weight Classifier	31	60.3%	0%
2	6 users, 2 days	Cost-based Classifier	11	0% (31.3%)	35.3%
3	4 users, 5 days	2-stage Classifier	29	6.5% (23.7%)	3.3%

TP=true positive, FP=false positive, FN=false negative

Table 1: Details of In-the-Wild Studies

2.3 Micro-Study Details

To realise the design goals and to gain detailed understanding of eating gestures, initial controlled studies were conducted with 21 participants (8 females, 13 males), who were employed in our research lab. The age range of the participants was between 24 to 35 years, and they belonged to 5 different nationalities. The participants contributed to a total of 135 eating episodes, where an episode is defined as the period between starting of a meal (after the purchase) and consuming the last spoonful. During the meal, the participant wore the watch on their dominant hand. A custom application running on the watch collected accelerometer, gyroscope and image frames, while an external observer video-recorded the meal (for ground truth labeling). The food items consumed by the participants included: rice (66 episodes), sandwich (20 episodes), pasta/noodles (29 episodes) and fruit pieces (15 episodes).

Initial Observations: We observed that there is wide variation in eating gestures for different food types considered. Eating episodes lasted anywhere between 51 seconds (fruits) to 19 minutes (rice), involving 6 (sandwich) to 54 (rice) separate hand-to-mouth gestures. Among these food items, we also observed from the videos that: (a) sandwiches/fruits presented the least number of distinct hand-to-mouth gestures (as users often held the items close to their mouth between successive bites), (b) "noodle/pasta" had high variability in the number of hand-to-mouth gestures mainly due to the use of forks vs. chopsticks, while the variation for "rice" is generally due to the individual eating speed and quantity consumed in each mouthful.

2.4 Real-World Studies & System Evolution

Several system-level choices in *Annapurna* occurred in an evolutionary fashion: an initial implementation was developed based on initial-studies (Section 2.3) and then deployed for an initial in-the-wild study. Lessons learnt from the study were then used to iteratively refine various system choices and parameters, via two additional in-the-wild studies. To better understand the evolution of each component, we provide details of the three user studies upfront, with a summary in Table 1. In each of these studies, participants manually recorded the ground truth. The eating activities recorded spanned a wide variety of environments and involved various types of food, eating modality and sitting position.

Study 1: 7 participants (4 females, 3 males; belonging to 3 nationalities) from our lab registered with *Annapurna*. They were provided with the watch (which they were instructed to wear in their dominant hand) and the phone. They were also asked to appropriately recharge the battery whenever it drained out. There was no requirement laid regarding meals to eat and places to eat. Other than this, the users were also asked to validate the accuracy of the system at the end of the day by logging into the journal.

By day 3 of the study we found that our gesture recognition system had high false positives, leading to rapid drainage of the smartwatch battery. Nonetheless, the participants used this version for 5 days, capturing a total of 31 eating episodes. This problem was traced to our use of a very lightweight classifier (chosen to ensure it could run on the watch) and the lack of robust real world data of a variety of non-eating activities. We then tried to deploy more complex classifiers (e.g., SVM, HMM), but found that they were too computationally demanding for the smartwatch. Consequently, we eventually switched to a cost-based classification approach (details in Section 3.4), where false-positives were more heavily penalized.

Study 2: We then redeployed an improvised cost-based classifier on 6 users (one of the original users dropped out) and evaluated it for 2 days. The new system significantly lowered the false positives in gesture recognition (eating gesture recogniser identified 5 false positive eating episodes; all of these were eventually filtered out (by the image filtering step). However, this classifier now exhibited higher false negative rate. We missed out 6 eating episodes over those 2 days. To subsequently tackle this issue, we then developed a two-stage eating detection classifier (details in Section 3), where a longer frame was used to identify potential eating gestures and a shorter frame confirmed if the gesture was indeed an eating gesture.

Study 3: The final refined version of the *Annapurna* client was tested on 4 (out of the original 7) users over another 5 day period. Using this study, we were finally able to demonstrate our target goal of achieving both low false positives and false negatives in real world conditions.

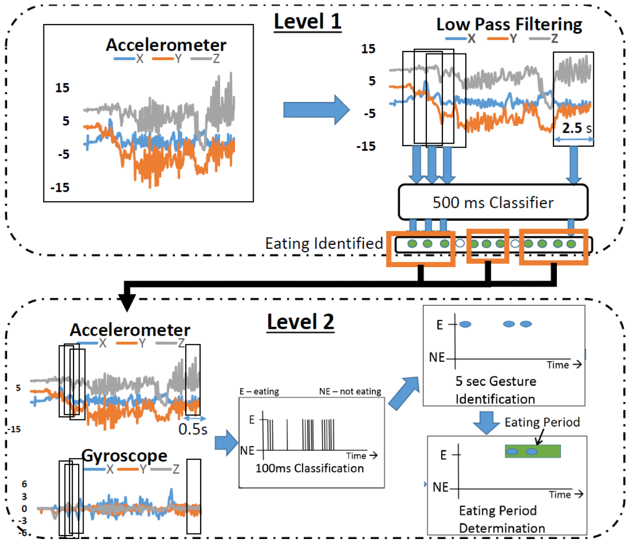


Figure 3: Recognising Eating Period. Level 1 frame width is 2500ms and outputs every 500ms. Level 2 frame width is 500ms and outputs every 100ms

3. DETECTING EATING GESTURES

Our overall design of the classifier for detecting eating gestures and episodes is shown in Figure 3. The entire system can be divided into four parts. We first describe the initial implementation of this classifier, and then describe the refinements that we made based on experiences gathered from real-world studies.

3.1 Feature Extraction and Classification

We extracted the raw accelerometer and gyroscope data from the eating episodes and manually labeled the hand-to-mouth gesture periods. From this data we found that an eating episode, on an average, has about 18 to 19 eating gestures. Our initial approach was to use features defined over short frames of 500 milliseconds for both accelerometer and gyroscope data. The small frame size is needed to trigger the camera reasonably in advance to get appropriate images. This approach is shown in the bottom part (Level 2) of Fig 3. The raw sensor data is partitioned into frames of length 500 msec (with 80% overlap between frames); a set of widely-used time and frequency domain features for the three axes of both accelerometer and gyroscope (mean, variance, covariance, correlation,entropy,energy – identical to features extracted in [15]) are then derived for each frame. From the features we built a *person-independent* classification models. Table 2 shows the accuracy, precision and recall of 10 fold cross validation for three commonly used classifiers. From the table we observe that both Decision Tree and Random Forest classifiers offer high classification accuracies. However, for our studies, we selected the Decision Tree classifier due to its lower computational complexity.

For a 500 msec window of sensor traces, we found that even during an eating gesture, two consecutive frames were not always classified as *eating*. Similarly, *non-eating* gestures (adjusting one's hair, raising the hand to wave at a friend, etc.) were classified as eating in several 500 ms windows. On average, our classifier's prediction indicated 337 transitions from *non-eating* to *eating*. This is much higher than the ground-truth (average of only 18-19 gestures), indicating the need of a second window to smoothen the noise.

Classifier	Accuracy	Precision	Recall
Decision Tree	96.63%	96.1%	96.5%
Random Forest	98.19%	97.1%	99%
SVM	85.66%	83.6%	87.1%

Table 2: Accuracy in identifying eating gestures

w	t (count)				
(sec)	10	20	30	40	50
2	-152.1	■	■	■	■
5	-4.2	-22.2	-3.4	■	■
10	48.3	35.7	34.3	35.9	33.9

Table 3: Gesture Prediction Error (%) for different window size(w), threshold (t).

3.2 Determining Length of an Eating Gesture

From the ground truth data we found that on average an eating gesture lasted for 3.1 seconds (Rice - 2.8 sec, Noodles - 3.7 sec, Sandwich 3.1 sec) where a gesture starts from the point the hand starts moving upwards and ends when the hand comes back to rest. To determine if a gesture determined by the 500 millisecond window was actually eating, we take a window(w) of past raw classifier outputs (obtained every 100ms) and compare the number of *eating* gestures identified by the classifier during this window with a threshold(t) value. If the total number of classifications in w is more than t, then we declare the window to be an eating gesture window. Table 3 shows the average error in determining the number of gestures (transitions from *notEating* to *eating*) in an episode, as a function of w and t. We computed $PredictionAccuracy = ((\Sigma GT - \Sigma P)/\Sigma GT) * 100$, where GT is the total number of eating gestures (ground truth) and P is the system-predicted gesture count. (A +ve value indicates that our system is under estimating, while a negative value indicates over-estimation.) From this table, we see the lowest values of error in gesture estimation are obtained for $w = 5$. A smaller value ($w = 2$sec) over-estimates the number of eating gestures, whereas an overly large window ($w = 10$sec) undercounts the number of eating gestures as it stays in the *eating* state for too long.

When we compared the estimation errors for different settings of w and t for individual food items (rice and noodles), we found that they are indeed different, due to the different eating styles. (In case of noodles, the user usually holds the hand near the mouth till she has consumed the entire strand of noodle.) However, even though t and w varied across different food items, the variation was modest enough to allow us to use $t = 10$ and $w = 5$ across food-types (i.e., for our gesture recognizer to be *food independent*).

3.3 Determining Eating Period

From the study, we observed that on average during a rice eating episode, an eating gesture occurred every ≈17 seconds. From the ground truth observation, we also saw that these gestures were not evenly distributed, but were rather bursty. On average, the first minute of the rice eating episode had ≈ 3 eating gestures. Hence, we decided to detect an *eating episode* only if our system detected at least 2 eating gestures within a minute.

	0	20	35	50	100
False Positives	36.6	18.9	12.6	8.6	6.7
False Negatives	3.5	8.9	17.4	37.1	55.3

Table 4: Error Rates for Different Cost Parameters

3.4 Refining the Classifier

Step 1—Building a Cost-Sensitive Classifier: When the base classifier (described above) was applied in User Study 1, it resulted in a high positive rate (see Table 1). This triggered detection of many false eating episodes and drained the battery rapidly by turning on the camera needlessly. To tackle this problem, we then increased the cost of false-positive misclassification in the training phase, thereby building a cost-sensitive classifier. However, from in-the-wild study 2 (Table 1), we found that we now suffered from unacceptably high false-negatives (missing several real eating episodes).

Step 2–Cost-Sensitive, Two-stage Classifier: The following improvements were needed for version 3: (a) We needed to determine the optimum cost for the classifier that provides the best trade-off between false positives and false negatives, and (b) We also needed an additional pre-classifier, that works on large frame size, to reduce the false-positives.

To get the optimum cost parameter, we first built five J48 classification models for 5 different cost settings – (0, 20, 35, 50, 100). Additionally, we acquired day-long regular life-style sensor traces of non-eating activities from 3 participants (The participants were asked to remove their watches when they are eating and wear them at other times.). For the models with different cost parameter settings, the false negative rate ($FN/(FN+TP)$), was determined from cross-validation on the micro-study training dataset itself. To evaluate the false-positive rate ($FP/(FP+TN)$), we used the day long traces of non-eating data (from these 3 participants). Table 4 provides the false-positive and false-negative rates for different values of cost parameter. When there is no cost, the FN rate is low, meaning we will not miss many eating gestures. However, the FP rate on real-life trace is very high (36.8%). For a cost of 100, the FP rate on the real-life trace is very low (6.7%), but the FN rate for eating is also very high (55.25%), implying we will miss most of the eating gestures. From this table, we observed that a cost parameter of 35 provides a low value for both FP rate (12.6%) on the real-life trace and the FN rate (17.4%) for detecting eating gestures.

In addition, we observed that several false-positives were generated by "jerky movements" of the hand during regular activities such as gesticulating during interactions or repeated lifting of objects etc. While a small frame-duration of $500ms$ is needed for efficient, low-latency triggering of the camera, an additional larger-frame duration of $2.5sec$ was also needed to eliminate these other transient, short-lived gestures. Accordingly, we developed an additional classifier (Level 1, as shown in Figure 3) that uses a longer 2.5 sec second frame of accelerometer data alone, to first identify the *likely* eating episodes. As each eating episode is long-lived, this initial classifier can be used as a trigger for the fine-grained classifier (Level 2 in Figure 3) which works on the shorter 500 ms frames, additionally using the gyroscope readings also. Once the eating gesture is consistently detected in level 1 (for more than 10 frames within a minute), this triggers the cost-based classifier (described earlier) that operates on $500ms$ frames.

4. SYSTEM PERFORMANCE

The performance of the system for each study is presented in Table 1. In the table, numbers indicated in bracket in FP column indicates the false positives of the eating gesture recogniser, i.e. when a person was not eating, but the system determines otherwise, while the other number indicates the overall system's false positives. From the table, we can see that the false positive rates of study 1, 2 and 3 are 60%, 31% and 23% respectively. This indicates that choosing a cost sensitive classifier (study 2 and 3) indeed lowered the false positives of the system. Moreover, adding a two-stage classifier (study 3) not only improved the false-positive, but also filtered the "jerky" hand movements, thus improving false-negative rates of *Annapurna*. Since image was captured whenever eating period was determined, it was straight forward to remove the false positive episodes (no food image was present in these episodes). Thus, after filtering, for study 3, we could reduce the false positives from 23% to 6.5%, indicating that the system had reasonable performance in real-world settings.

In terms of false negatives, study 1 had the best performance since the system determined almost every hand movement as eating. In subsequent studies, we missed out on some eating episodes because of the cost associated with the classification and the system was careful in determining eating periods. However, overall the number of episodes missed in study 3 was just one.

Since an application can have its specific goal (e.g. not to miss any eating episode), the system parameters have to be tuned appropriately to meet the required goals in terms of acceptable FP and FN.

5. DISCUSSION

The studies demonstrated the possibility of deploying an eating gesture recognition system. However, for robustness, additional factors have to be considered.

Dominant Hand: In this paper we have assumed that the watch is worn on the dominant hand. This assumption is in-line with several recent works [7, 12]. To determine the validity of this assumption, we conducted a survey through Amazon MTurk, where we asked participants if they would wear the watch on their dominant hand. From the small set of responses (30 respondents), we found that 50% of 20 respondents who wear a smartwatch, wear it on the dominant hand. Furthermore, 70% of the watch wearers indicated that they would be willing to wear the watch in a dominant hand if it could create an automated food journal. Even though the number of respondents is small, the response towards wearing the watch on the dominant hand appears to be positive. Alternately, *Annapurna* could utilise data from other wearable devices (e.g. fitness bands) if they are worn on the dominant hand.

For our current studies we had identified that the 'hand-to-mouth' gesture provides a window of opportunity for capturing food plate images. In future, we plan to study the role of the non-dominant hand during an eating episode and determine if we can identify moments when images can be opportunistically captured.

Demographic Diversity: Our current studies have been validated on a small group of similar participants (age group, job profile). Since eating detection technique might be useful for other demographic groups (e.g. elderly, children), studying the characteristics of eating for these demographics will open a new set of challenges. Additionally, currently we have considered sit-and-eat meals. It will be interesting to study the challenges in identifying on-the-go eating detection (e.g. in child's diet monitoring).

Battery Life: Currently we have applied energy saving techniques in both gesture identification (cheaper sensor (accelerometer) turns on the more expensive sensors(gyro and camera)) as well as in the image capturing module. With these techniques, the watch has a battery life of ≈ 12 hours. Even though the improvement in battery life is significantly higher than continuous video capture (battery life is ≈ 80 mins), more innovative approaches (e.g. duty cycling) have to be considered to further improve the battery life to atleast one day.

6. RELATED WORK

Wearable based Gesture Recognition: Numerous researchers have focused on using the smartwatch to determine hand based gestures. Work such as [5] uses a smartwatch to determine driving behavior, while the authors in [7] have demonstrated the possibility of determining smoking gestures. Similarly, authors in [10] and [13] have used the smartwatch to identify various key-press patterns. All these studies have demonstrated the possibility of utilising the inertial sensor data from a smartwatch to recognise hand gestures. Alternately, work such as [14] uses subtle movement in the smartwatch to determine finger gestures. Our work is similar to these works as we also use sensor data from the smartwatch to identify a specific gesture – *eating*.

Eating Detection using Inertial Sensors: While the authors in [1] demonstrated the possibility of determining the eating gesture using a custom hardware with inertial sensors, the authors in [2] determined the amount of food consumed by an individual based on the spoon count. More recently, studies such as [11] and [12] have demonstrated the possibility of utilising an off the shelf device to determine eating gestures. Alternately, the authors in [6] demonstrated the possibility of multimodal sensing to identify eating gestures as well as the food consumed.

Eating detection through Non-Inertial Sensors: Researcher have used sound to determine eating – e.g. in [8], the authors utilized a neck-attached microphone attached to identify various body sounds, including eating-related ones. The authors in [9] used the camera to continuously captured images to identify food items consumed, while in [4], the authors utilised special hardware to opportunistically capture the images. Our work relies on inertial sensing, and employs only commodity devices.

7. CONCLUSION

In this paper, we described our experiences in implementing *Annapurna's* eating gesture recogniser. We described various design choices to ensure acceptable real world performance (FP and FN rates of 6.5% and 3.3% respectively). Key innovations that we reported in this paper included: (a) a cost-weighted classifier to filter real-world similar non-gestures and (b) a 2-tier classifier to capture the diversity of eating styles and gestures.

8. REFERENCES

[1] Amft, O., Junker, H., and Troster, G. Detection of eating and drinking arm gestures using inertial body-worn sensors. *Wearable Computers. International Symposium on*, 2005.

[2] Dong, Y., Hoover, A., Scisco, J., and Muth, E. A new method for measuring meal intake in humans via wrist motion tracking. *Applied psychophysiology and biofeedback*, 2012.

[3] Lee, J., Banerjee, A., and Gupta, S. K. Mt-diet: Automated smartphone based diet assessment with infrared images. *Pervasive Computing and Communications (PerCom)*, 2016.

[4] Liu, J., Johns, E., Atallah, L., Pettitt, C., Lo, B., Frost, G., and Yang, G.-Z. An intelligent food-intake monitoring system using wearable sensors. *Wearable and Implantable Body Sensor Networks (BSN), 9th International Conference on*, 2012.

[5] Liu, L., Karatas, C., Li, H., Tan, S., Gruteser, M., Yang, J., Chen, Y., and Martin, R. P. Toward detection of unsafe driving with wearables. *Proceedings of the 2015 Workshop on Wearable Systems and Applications*, 2015.

[6] Mirtchouk, M., Merck, C., and Kleinberg, S. Automated estimation of food type and amount consumed from body-worn audio and motion sensors. *Proceedings of the ACM Conference on Pervasive and Ubiquitous Computing*, 2016.

[7] Parate, A., Chiu, M.-C., Chadowitz, C., Ganesan, D., and Kalogerakis, E. Risq: Recognizing smoking gestures with inertial sensors on a wristband. *12th International Conference on Mobile systems, applications, and services*, 2014.

[8] Rahman, T. et al. Bodybeat: A mobile system for sensing non-speech body sounds. *International Conference on Mobile Systems, Applications, and Services*, 2014.

[9] Reddy, S., Parker, A., Hyman, J., Burke, J., Estrin, D., and Hansen, M. Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype. *4th workshop on Embedded networked sensors*, 2007.

[10] Sen, S., Grover, K., Subbaraju, V., and Misra, A. Inferring smartphone keypress via smartwatch inertial sensing. *Pervasive Computing and Communication Workshops (PerCom Workshops), IEEE International Conference on*, 2017.

[11] Sen, S., Subbaraju, V., Misra, A., Balan, R. K., and Lee, Y. The case for smartwatch-based diet monitoring. *Pervasive Computing and Communication Workshops (PerCom Workshops), IEEE International Conference on*, 2015.

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

[13] Wang, C., Guo, X., Wang, Y., Chen, Y., and Liu, B. Friend or foe?: Your wearable devices reveal your personal pin. *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, 2016.

[14] Xu, C., Pathak, P. H., and Mohapatra, P. Finger-writing with smartwatch: A case for finger and hand gesture recognition using smartwatch. *16th International Workshop on Mobile Computing Systems and Applications*, 2015.

[15] Yan, Z., Subbaraju, V., Chakraborty, D., Misra, A., and Aberer, K. Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. *Wearable Computers (ISWC), 16th International Symposium on*, 2012.