EarComp 2019

1st International Workshop on Earable Computing

In conjunction with UbiComp 2019
September 10 , London, UK

TABLE OF CONTENTS

Towards Respiration Rate Monitoring Using an In-Ear Headphone Inertial Measurement Unit

Tobias Röddiger, Daniel Wolffram, David Laubenstein*, Matthias Budde, Michael Beigl
{roeddiger,wolffram,budde,beigl}@teco.edu,*{david.laubenstein}@student.kit.edu
Karlsruhe Institute of Technology
Karlsruhe, Germany

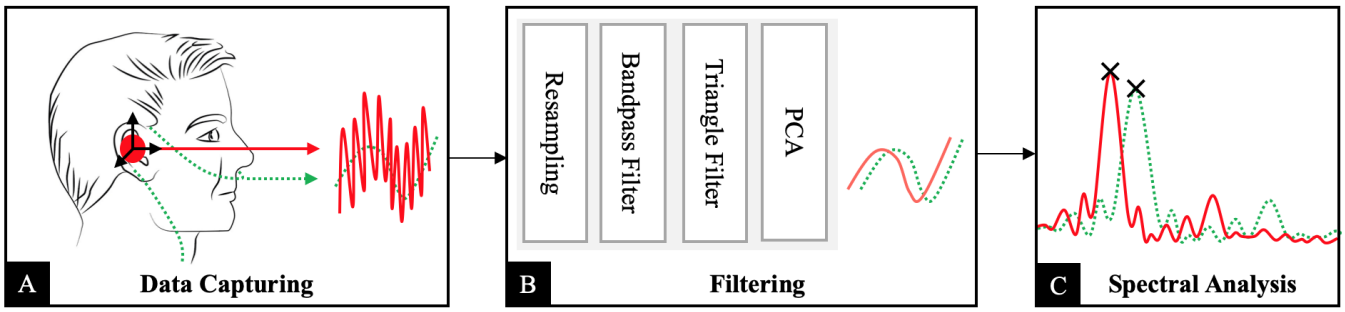


Figure 1: To measure respiration with an in-ear headphone IMU, we capture 3D acceleration and gyroscope data at 50 Hz (A). We record the ground truth with a pressure transducer. Data is processed as illustrated in (B): we split the data into 20-second windows and interpolate using cubic splines, resampling at 256 Hz. Our pipeline discards windows with too much movement. We apply a Butterworth bandpass filter to remove noise and a triangle filter for further smoothening without loss of timing information. Finally, we use FFT with zero padding in (C) and compute the maximum to calculate the respiratory rate.

ABSTRACT

State-of-the-art respiration tracking devices require specialized equipment, making them impractical for every day at-home respiration sensing. In this paper, we present the first system for sensing respiratory rates using in-ear headphone inertial measurement units (IMU). The approach is based on technology already available in commodity devices: the *eSense* headphones. Our processing pipeline combines several existing approaches to clean noisy data and calculate respiratory rates on 20-second windows. In a study with twelve participants, we compare accelerometer and gyroscope based sensing and employ pressure-based measurement with nasal cannulas as ground truth. Our results indicate a mean absolute error of 2.62 CPM (acc) and 2.55 CPM (gyro). This overall accuracy is comparable to previous approaches using accelerometer-based sensing, but we observe a higher relative error for the gyroscope. In contrast to related work using other sensor positions, we can not report significant differences between the two modalities or the three postures standing, sitting, and lying on the back (supine). However, in general, performance varies drastically between participants.

1st Int. Workshop on Earable Computing, September 10, 2019, London, UK

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Applied computing** → *Consumer health.*

KEYWORDS

In-Ear Headphones, Respiration, Monitoring, IMU, Breathing

ACM Reference Format:
Tobias Röddiger, Daniel Wolffram, David Laubenstein*, Matthias Budde, Michael Beigl. 2019. Towards Respiration Rate Monitoring Using an In-Ear Headphone Inertial Measurement Unit. In *1st International Workshop on Earable Computing, September 10, 2019, London, United Kingdom.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

When tracking respiration rates in day-to-day scenarios, respiratory inductance plethysmography is the current state-of-the-art. A belt straps around the user's chest and abdominal wall to measure the expansion while breathing in and out [21]. Such devices are specialized and expensive equipment and not suitable for everyday use. A different method, applied for instance in sleep labs and under medical conditions, uses nasal cannulas made from plastic tubes which redirect the airflow to pressure transducers [15]. These tubes are uncomfortable to wear as they are placed inside the nostrils and are unhygienic when used multiple times or with different users. As an alternative, we investigate respiratory monitoring based on a 6-axis inertial measurement unit (IMU) embedded into a standard in-ear headphone form factor. This technology is potentially accessible to a broad set of users, as already today, earphones with

integrated IMUs are commercially available (like e.g., the *Apple AirPods*).

Embedding respiratory sensing into headphones opens up a set of use cases where auditory feedback couples to breathing. For example, Harris et al. suggest that auditory biofeedback can enable the control of the respiration rate of users [5], which can help with stress management or support guided meditations [17]. We could also detect the interruption of breathing during sleep (apnea) and alert the user after a defined time threshold. Such scenarios naturally fit our sensor setup because the user maintains a steady position which avoids movement artifacts, and the headphones seamlessly integrate wit providing audio feedback.

In this paper, we propose the use of headphones equipped with a 6-axis IMU (accelerometer and gyroscope). We present the working principle of our system and introduce a data processing pipeline. We evaluated our system in a lab study with twelve participants and compared between standing, sitting, and lying on the back (supine) as well as accelerometer and gyroscope based tracking. Our results indicate good outcomes for a subpopulation of participants.

2 BACKGROUND AND RELATED WORK

Previous research proposes a broad set of alternatives to the formerly mentioned state-of-the-art for respiration rate tracking.

Systems based on UWB, WiFi, or vision [1, 12, 19] have significant advantages because they are not attached to the user. However, they require specialized setups which can be complicated or might have problems with other people present in the room. Further techniques which do not require nasal cannulas to measure respiration from human breath include gas sensors that measure, e.g., volatile organic compounds [16] or humidity [13]. They still require to be placed in the air stream or have to be attached close to the area around mouth and nose.

Acceleration and gyroscope data have significant advantages in terms of cost and unobtrusiveness because many modern devices already come equipped with these inexpensive sensors (< 5$, single quantity). The idea to use the IMU's data is not new, and others have shown that the underlying principle does work. It has been implemented using chest belts [2] or smartwatches [3, 18]. Another approach which is particularly relevant because it is head-worn uses *Google Glass* smart glasses [7]. However, this is impractical for scenarios where glasses can not be comfortably worn, e.g., when lying on the side. Additionally, it remained questionable to us if a sensor plugged into the ear canal is capable of achieving similar results. Nevertheless, we can use their results to add context to our system's performance and draw additional conclusions. Existing work has also explored the option to support meditation sessions with breathing monitored in real-time using a so-called breathing model derived from a smartwatch's IMU [4].

Tracking respiration rates with a headset has been explored in [14] but their method requires placing the microphone of the headset under users nose. In [11] a microphone is positioned in the ear-canal using in-ear to record und interpret breathing sounds. It was tested with 25 subjects and yielded 2.7 breathing cycles per minute (CPM) absolute mean error for quiet environments, and they report problems with increasing background noises. In [6], an accelerometer-equipped device wraps around the user's ear to measure respiration. They only evaluate the device with a single participant and do not consider gyroscope data as well as an in-ear form factor.

3 WORKING PRINCIPLE

3.1 System Design

Figure 2: The Nokia Bell Labs eSense headphones [9] (left) connect via Bluetooth and transfer gyroscope and accelerometer data to the smartphone app (right).

We leverage the *eSense* platform [9], which has been kindly provided to us by *Nokia Bell Labs*. It comes equipped with a six-axis IMU in its left earbud and connects via Bluetooth Low Energy (BLE). We record the x, y and z angular velocity in deg/s using the built-in gyroscope and the x, y and z acceleration in m/s^2 as illustrated in Figure 2. We sample at the maximum frequency of 50 Hz and do not use the integrated low pass filter settings. We do not record other information provided by the platform (e.g., microphone).

We have implemented a mobile application in Swift for iOS, which connects to the eSense earbuds. The data is stored locally on the phone, and timestamps are taken on a rolling basis as the Bluetooth packages arrive. To transfer the data to a computer, we export it to a CSV file and send it using any of the options provided natively by the operating systems' sharing options (in our case, *AirDrop*). Finally, we feed the files into our processing pipeline.

3.2 Data Processing Pipeline

To compute the respiration rate, we use the same steps independent from gyroscope or accelerometer. We expand upon the approach proposed in [7]. We apply steps (1), (2), (4), and (5) to each axis. Additionally, steps (3) and (5) remove motion sensitivity.

(1) To remove signal shifts and trends, a moving average window of 3 samples is subtracted from each dimension. Additionally, we apply an averaging filter with a window size of 2 seconds to each of the components, corresponding to one respiration cycle at the maximum breathing rate (30 breaths per minute).
(2) To inflate our data, we apply a cubic spline interpolation and resample the resulting signal at 256 Hz. As there are small variations in timestamps due to Bluetooth latency, this also helps to create equidistant samples.
(3) Similar to [18] we discard windows if movement is too high. If 3% or more of all accelerometer data points are above our threshold of 10 m/s^2 the entire sequence is sorted out. Additionally, we apply hard thresholding for samples ±2SD.
(4) We apply a bandpass Butterworth filter of order four and cut-off frequencies of 0.1 Hz and 0.5 Hz, which removes noise and is equivalent to 6 to 30 breath cycles per minute (CPM).

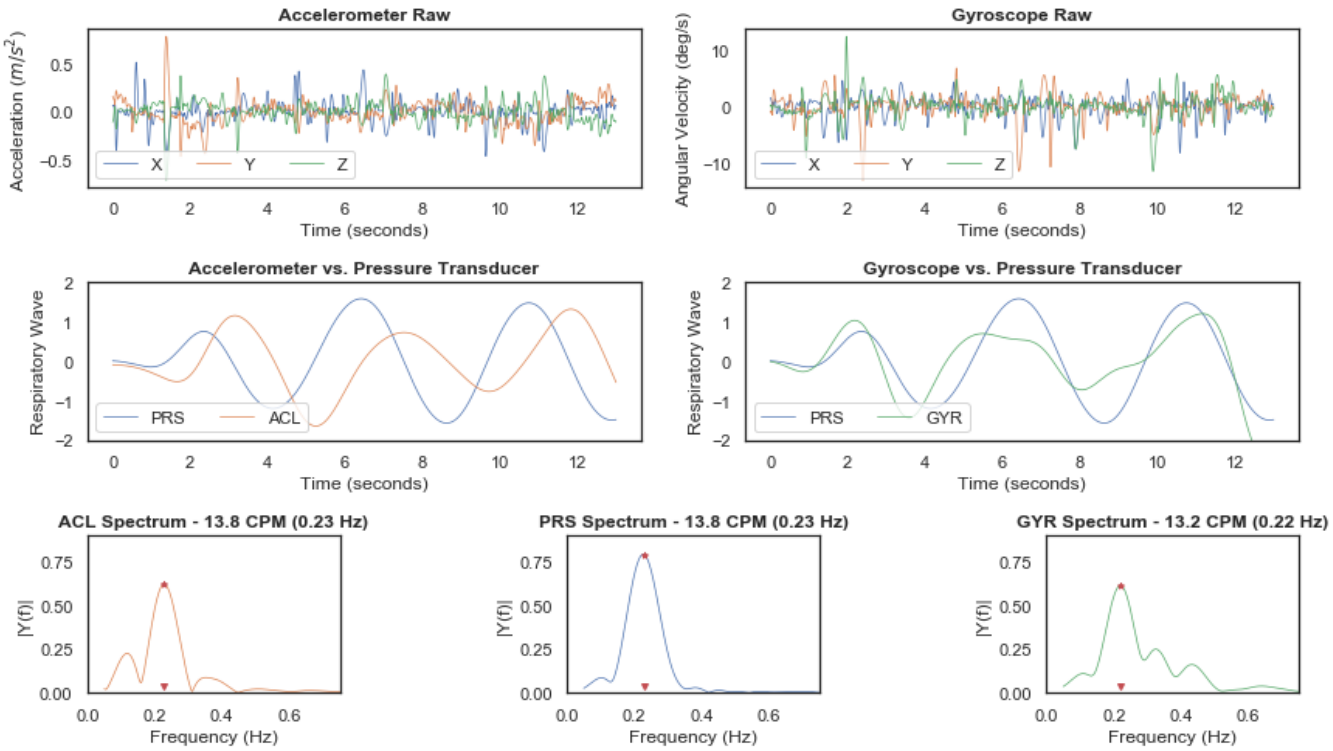


Figure 3: The graph in the top left corner shows the raw acceleration signal of the X-, Y- and Z-axis, the graph in the top right the gyroscope data of the X-, Y- and Z-axis. The graph shown at the center-left displays the filtered acceleration signal compared to the ground truth and the center-right the filtered gyroscope signal compared to the ground truth. The three graphs in the bottom show the spectrum of the processed accelerometer signal (left), the ground truth pressure signal (center) and the gyroscope (right).

(5) To further smoothen the signals while retaining the peak positions, we apply a triangle filter with a width of 2 seconds as described in [4].
(6) To make the results independent from changes on different axes for different postures, we perform a principal component analysis (PCA).
(7) We perform a spectral analysis of each principal component using a Fast Fourier Transformation (FFT) with zero-padding and compute the maximum peak and its magnitude for each component. We then report the frequency corresponding to the peak with the highest magnitude as respiration frequency that we can convert to CPM.

Figure 3 displays the raw signals captured from the accelerometer (left) and gyroscope (right) of one of our study participants sitting. The two graphs indicate how noisy the initial data signals are along all axes. As the user breathes, the accelerometer signal visibly oscillates around zero on the Y-axis and the gyroscope on the Z-axis. The second row compares a normalized ground truth signal captured using nasal cannulas hooked to a pressure transducer (PRS) with our filtered signal. The bottom row displays the three different spectra computed from the respective sensor signals. A red star indicates the maximum in each spectrum, which illustrates a minimal error between the three different modalities.

4 EVALUATION

4.1 Study Design

To evaluate our system, we recruited twelve participants (two female, ten male) between the ages of 21 and 39 (mean age 26) for a lab study. The mean height was 179 cm and weight 81 kg. We did not pay participants. Our experiment was conducted in a room with a couch to lie on and a stable chair with armrests for sitting down. We placed both *eSense* earbuds [9] into the participant's ears (left earbud equipped with the IMU) and hooked them up to nasal cannulas as ground truth. No audio was played during the study.

Figure 4: Participant wearing nasal cannulas (red, left circle) and the eSense headphones (blue, right circle).

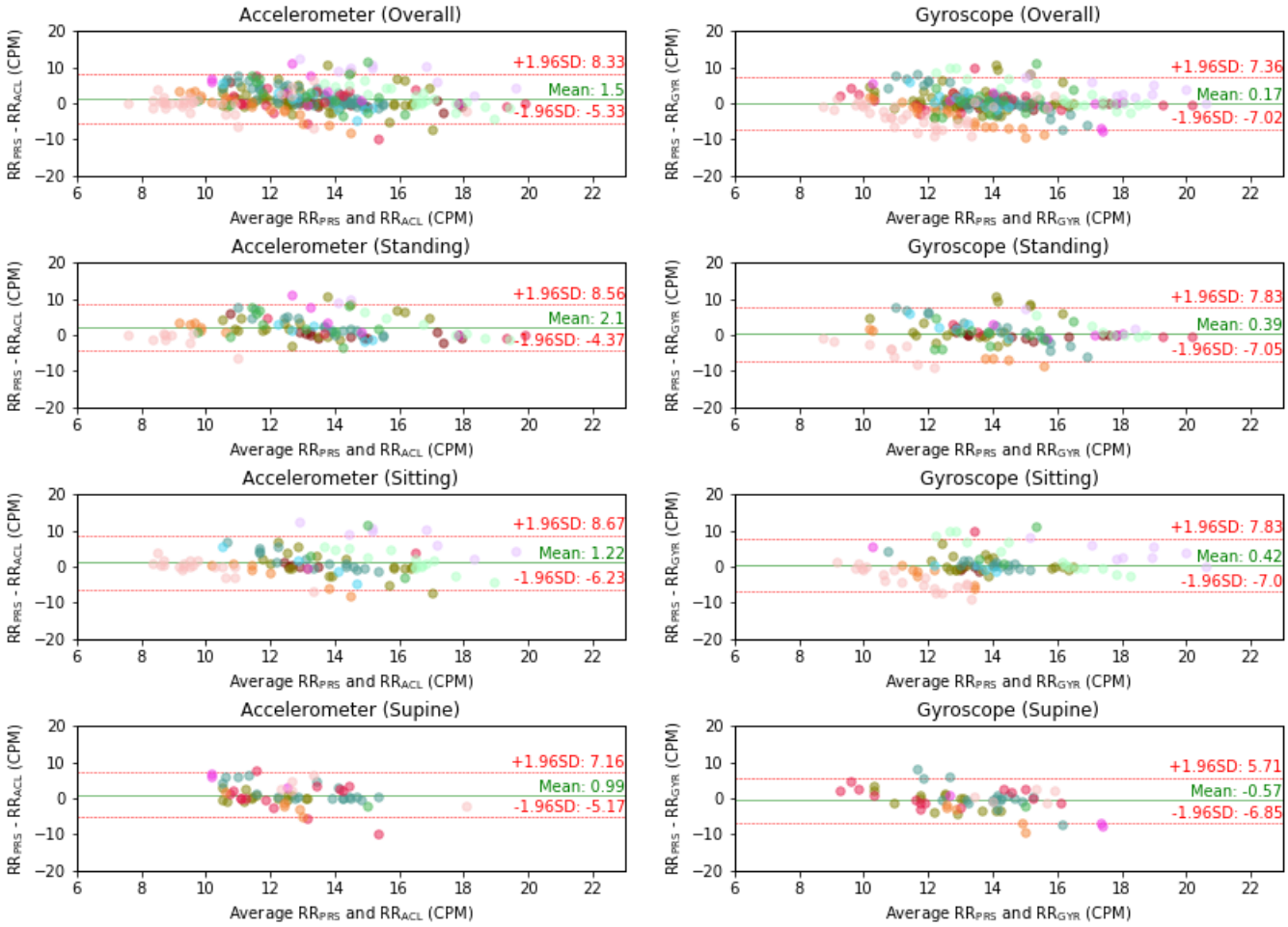


Figure 5: Shows Bland-Altman plots for the accelerometer (left) and gyroscope (right) with an aggregated graph for the overall performance and for the postures standing, sitting and lying on the back. Per-User results indicated in different colors.

Table 1: Respiration rate system performance in cycles-per-minute (CPM).

Sensor	MAE	SD	RMSE
Accelerometer	2.62	2.74	3.79
Gyroscope	2.55	2.63	3.67

Table 2: Comparison between modalities and postures (MAE / SD) in cycles-per-minute (CPM).

Sensor	Standing	Sitting	Supine
Accelerometer	3.15 / 2.74	3.10 / 2.80	2.56 / 2.19
Gyroscope	2.45 / 2.22	2.74 / 2.64	2.68 / 2.00

During the first phase of our evaluation, each participant was asked to breathe normally for one minute each in three different postures (standing, sitting, and lying) while otherwise keeping as still as possible. This step was followed by a second phase, in which we then asked them to perform a short 30-second jumping jacks session before each of the three postures, which we anticipated to result in a more dynamic dataset. After performing the activity, we again recorded respiration data for one minute for every posture.

We used the Williams design generalized Latin squares [20] to balance for first-order carryover effects introduced by a potentially unnatural breathing behavior when asked to breathe on the spot. For the three different postures, this resulted in six different sequences, which we assigned to participants in a round-robin fashion according to their arrival time. The sequence for the first and second phase of the evaluation was identical within each session. After completing these tasks, participants were asked to fill in a short questionnaire, which included demographic questions (sex, age, weight, height) as well as a question inquiring whether participants felt that they had breathed naturally and space for free-text feedback.

The respiration ground truth was collected using a custom made monitoring device: We wired a *RedBear BLE Nano v2* to a pressure

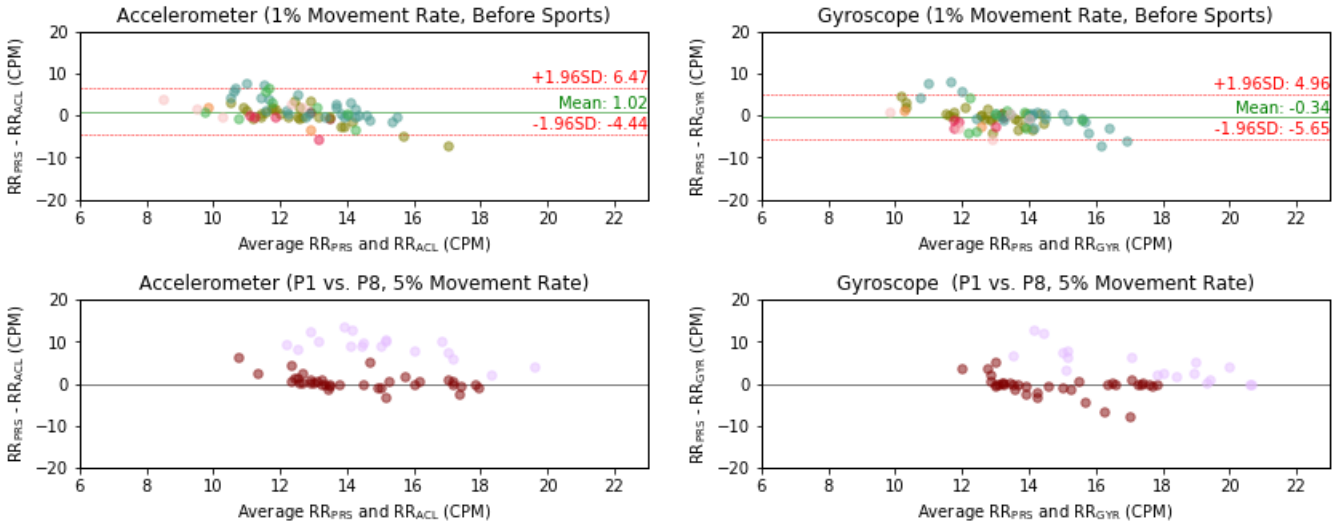


Figure 6: The two diagrams in the first row above indicate how strictly discarding segments with movement artifacts and of restful users increases the accuracy. The diagrams in the second row compare the data of two participants, whereas P1 achieves a much lower mean error than P8 even at a higher movement threshold of 5%. Per-User results indicated in different colors.

transducer that connects to nasal cannulas (see Figure 4). The device samples pressure data at a frequency of 50 Hz. We filter the signal with the same data processing pipeline as described in 3.2, except we do not apply a PCA. It also connects to our mobile application, which we installed on an *Apple iPhone X* for our study. Similarly to the acceleration and gyroscope data, we use the arrival time of the Bluetooth packages on the phone as timestamps, which makes it easy to synchronize the data afterward. We also attached the device on the user as displayed in Figure 4.

4.2 Results

In total, we collected 72 minutes of breathing data. We shift a sliding window at an interval of 5 seconds over every one-minute data frame. This process yields 669 twenty-second breathing sequences. After removing the ones with too many movement artifacts, 253 remain. Ground truth respiration rates range from 7.6 to 22 in cycles-per-minute (CPM) for those sequences. To evaluate the agreement between our approach and the ground truth measurement, we utilize Bland-Altman plots shown in Figure 5. In most settings, the observed differences are centered around zero and show no significant bias, also observable from the displayed mean error. The plots also show the limits of agreement (interval between +1.96SD and -1.96SD) that contain 95% of the measured differences. Additionally, we have computed several metrics for ease of comparison, namely the mean absolute error, its standard deviation, and the root mean squared error shown in Table 1. We further broke it down by body posture in Table 2. Overall, the performance of gyroscope is similar to the accelerometer but varies between postures. We achieve the best results for the accelerometer in the supine position, followed by similar results for sitting and standing. For gyroscope, we achieve comparable results for all three postures.

5 DISCUSSION

Generally, we can observe that our method is highly sensitive to motion artifacts. The first row in Figure 6 shows that we achieve better results after setting the motion threshold to 1% and limiting the dataset to non-aroused participants. Introducing this limitation reduces the MAE to 2.09 CPM for the accelerometer and to 1.90 CPM for the gyroscope. Additionally, we see significant differences between subjects. For example, the second row in Figure 6 shows that even after raising the motion threshold to 5% participant P1 has much better results than P8 (MAE 1.21 ACC / 1.45 GYR vs. MAE 8.97 ACC / 4.58 GYR). We do not know what causes these differences; however, bad fitting of the earplugs or differences in pose and anatomy could be a reason.

5.1 Comparison with Related Work

In Table 3, we compare our results to related work. Hernandez et al. [7] have evaluated smart glasses and also smartwatches [8] for the same three postures as we did. Compared to [7] we do not observe significant differences between gyroscope and accelerometer. Overall, the head seems to be a less suitable position for tracking respiration rates than e.g., the wrist. We yield higher error rates than [7, 18], especially for the gyroscope. The root causes for differences between the two head-worn devices are unclear.

Table 3: Performance (MAE / SD) in cycles-per-minute (CPM) of our system compared to related work.

Sensor	In-Ear	Glasses [7]	Watch [8]
Accelerometer	2.62 / 2.74	2.29 / 3.43	0.92 / 2.20
Gyroscope	2.55 / 2.63	1.39 / 2.27	0.38 / 1.19

Comparing the different poses, we have similar results to [7] for the accelerometer in the standing posture; however, perform worse for sitting and lying down. Additionally, the gyroscope's performance for standing is comparable, but we report higher errors for the sitting and supine position.

According to [10], the thoracic spine moves back and forth, whereas the spine moves up and down while breathing. The resulting motion of the head could be different when measuring above the eye compared to the ear in different poses and on varying head positions. We further theorize that the differences could be caused by a dampening effect of the ear plugs' flexible caps which might absorb motion.

5.2 Limitations

For our evaluation, we measured nasal respiration using a pressure transducer. After performing the physiologically straining task of jumping jacks, several participants reported the urge to breathe through the mouth afterward. We did not limit them to nose breathing before the study, but several participants reported that they "felt forced to not breathe through the mouth" (P2). An FDA-cleared chest belt based on respiratory inductance plethysmography is a more suitable methodology, which could support a more natural breathing behavior and therefore positively affect results. To identify relationships between the fit of the earplugs and respiration rate estimation accuracy, we suggest measuring ear sizes and film participants in future studies. Additionally, our experimental environment left things to explore visually (e.g., posters), which could result in additional motion artifacts.

6 CONCLUSION AND FUTURE WORK

In this paper, we explored the feasibility to use in-ear headphones for tracking respiratory rates and compared our results to related work. We have evaluated a data processing pipeline which combines multiple factors from previous work to fit an in-ear headphone use case. We compared accelerometer and gyroscope data and present results for the three different postures standing, sitting and lying on the back (supine). Our approach was validated by comparing measurements with ground truth data from nasal cannulas connected to a pressure transducer. In general, our results suggest that the ear is a less suitable position for measuring respiratory rates than, e.g., on the wrist. Overall, our solution has a high sensitivity to small motion artifacts. Nevertheless, we achieve stable performances for a subpopulation of participants. T he underlying root causes for those differences remain unclear at this point and will be investigated in future research.

To explore the inaccuracies between subjects, we suggest to investigate the potentially loose attachment of in-ear headphones further and examine possible dampening effects created by the headphone's cushion. We suggest using different sizes of ear canal caps depending on the user's ear size. We also propose a comparison study between the ear and above eye positions for respiratory rate tracking, which could reveal further insights that cause the performance differences discovered in this paper. Finally, a more advanced approach fusing accelerometer and gyroscope data and even microphone signals are likely to yield better results.

REFERENCES

[1] Heba Abdelnasser, Khaled A Harras, and Moustafa Youssef. 2015. UbiBreathe: A ubiquitous non-invasive WiFi-based breathing estimator. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 277–286.

[2] Andrew Bates, Martin J Ling, Janek Mann, and DK Arvind. 2010. Respiratory rate and flow waveform estimation from tri-axial accelerometer data. In *2010 International Conference on Body Sensor Networks*. IEEE, 144–150.

[3] Marian Haescher, Denys JC Matthies, John Trimpop, and Bodo Urban. 2016. SeismoTracker: upgrade any smart wearable to enable a sensing of heart rate, respiration rate, and microvibrations. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 2209–2216.

[4] Tian Hao, Chongguang Bi, Guoliang Xing, Roxane Chan, and Linlin Tu. 2017. Mindfulwatch: A smartwatch-based system for real-time respiration monitoring during meditation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 57.

[5] Jason Harris, Sarah Vance, Odair Fernandes, Avinash Parnandi, and Ricardo Gutierrez-Osuna. 2014. Sonic respiration: controlling respiration rate through auditory biofeedback. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2383–2388.

[6] David Da He. 2013. *A wearable heart monitor at the ear using ballistocardiogram (BCG) and electrocardiogram (ECG) with a nanowatt ECG heartbeat detection circuit*. Ph.D. Dissertation. Massachusetts Institute of Technology.

[7] Javier Hernandez, Yin Li, James M Rehg, and Rosalind W Picard. 2015. Cardiac and Respiratory Parameter Estimation Using Head-mounted Motion-sensitive Sensors. *EAI Endorsed Trans. Pervasive Health Technol.* 1, 1 (2015), e2.

[8] Javier Hernandez, Daniel McDuff, and Rosalind W Picard. 2015. Biowatch: estimation of heart and breathing rates from wrist motions. In *2015 9th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth)*. IEEE, 169–176.

[9] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (2018), 83–89.

[10] JCY Leong, WW Lu, KDK Luk, and EM Karlberg. 1999. Kinematics of the chest cage and spine during breathing in healthy individuals and in patients with adolescent idiopathic scoliosis. *Spine* 24, 13 (1999), 1310.

[11] Alexis Martin and Jérémie Voix. 2017. In-ear audio wearable: Measurement of heart and breathing rates for health and safety monitoring. *IEEE Transactions on Biomedical Engineering* 65, 6 (2017), 1256–1263.

[12] Manuel Martinez and Rainer Stiefelhagen. 2012. Breath rate monitoring during sleep using near-IR imagery and PCA. In *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 3472–3475.

[13] Umesha Mogera, Abhay A Sagade, Subi J George, and Giridhar U Kulkarni. 2014. Ultrafast response humidity sensor using supramolecular nanofibre and its application in monitoring breath humidity and flow. *Scientific reports* 4 (2014), 4103.

[14] Yunyoung Nam, Bersain A Reyes, and Ki H Chon. 2015. Estimation of respiratory rates using the built-in microphone of a smartphone or headset. *IEEE journal of biomedical and health informatics* 20, 6 (2015), 1493–1501.

[15] Robert G Norman, Muhammed M Ahmed, Joyce A Walsleben, and David M Rapoport. 1997. Detection of respiratory events during NPSG: nasal cannula/pressure sensor versus thermistor. *Sleep* 20, 12 (1997), 1175–1184.

[16] Tobias Röddiger, Michael Beigl, Marcel Köpke, and Matthias Budde. 2018. VOCNEA: sleep apnea and hypopnea detection using a novel tiny gas sensor. In *Proceedings of the 2018 ACM International Symposium on Wearable Computers*. ACM, 226–227.

[17] Ameneh Shamekhi and Timothy Bickmore. 2018. Breathe Deep: A Breath-Sensitive Interactive Meditation Coach. In *Proceedings of the 12th EAI International Conference on Pervasive Computing Technologies for Healthcare*. ACM, 108–117.

[18] Xiao Sun, Li Qiu, Yibo Wu, Yeming Tang, and Guohong Cao. 2017. Sleepmonitor: Monitoring respiratory rate and body position during sleep using smartwatch. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 104.

[19] Swaroop Venkatesh, Christopher R Anderson, Natalia V Rivera, and R Michael Buehrer. 2005. Implementation and analysis of respiration-rate estimation using impulse-based UWB. In *MILCOM 2005-2005 IEEE Military Communications Conference*. IEEE, 3314–3320.

[20] EJ Williams. 1949. Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Chemistry* 2, 2 (1949), 149–168.

[21] Jonathan D Witt, Jason RKO Fisher, Jordan A Guenette, Krystie A Cheong, Brock J Wilson, and A William Sheel. 2006. Measurement of exercise ventilation by a portable respiratory inductive plethysmograph. *Respiratory physiology & neurobiology* 154, 3 (2006), 389–395.

Head Motion Tracking Through in-Ear Wearables

Andrea Ferlini
University of Cambridge
af679@cam.ac.uk

Alessandro Montanari
Nokia Bell Labs Cambridge
alessandro.montanari@nokia-bell-labs.com

Cecilia Mascolo
University of Cambridge
cm542@cam.ac.uk

Robert Harle
University of Cambridge
rkh23@cam.ac.uk

ABSTRACT

Head tracking is a fundamental component in visual attention detection, which, in turn, can improve the state of the art of hearing aid devices. A multitude of wearable devices for the ear (so called earables) exist. Current devices lack a magnetometer which, as we will show, represents a big challenge when one tries to use them for accurate head tracking.

In this work we evaluate the performance of eSense, a representative earable device, to track head rotations. By leveraging two different streams (one per earbud) of inertial data (from the accelerometer and the gyroscope), we achieve an accuracy up to a few degrees. We further investigate the interference generated by a magnetometer in an earable to understand the barriers to its use in these types of devices.

1 INTRODUCTION

As of 2019, more than 466 million people are suffering from hearing impairments[1] [1]. By 2050, the World Health Organization forecasts that over 900 million people will be affected by hearing disabilities [1]. Subjects with hearing impairments (and non-impaired, too) struggle to identify and isolate the source of a sound. This is particularly true in social situations. Crowded places, where multiple conversations, involving several speakers happen at the same time, result in the phenomenon known as the *Cocktail Party Problem* [11].

Earables, like the eSense platform we evaluate in this paper, have an enormous and mostly unexploited potential. For instance, if adopted as hearing-aids, earables could be used both as sensors and actuators. Ears represent an extremely good vantage point to sense those behaviours that could be exploited to improve the performance of a hearing-aid (e.g. gaze tracking, head movements, etc.). Previous studies, especially from the medical community, highlight the importance of using directional hearing-aids leveraging the visual attention of the user [5, 14]. For example, Favre-Félix et al. [5] use electrooculography data (EOG) to characterize the visual attention of the patients. Their study shows how steering an hearing-aid by using EOG leads to better performance on the sentence correctness score. However, because of the challenging signal processing required when dealing with it, the authors suggest that EOG may not be the best enabler to steer hearing-aids with a sufficiently high degree of precision. Head movements are closely linked to eye-movements [5], and therefore they are considered a good proxy to sense visual attention, too.

Inertial motion tracking is a well known and studied problem. Yet, due to the lack of a reference point to re-calibrate the sensors, and to estimate the 3D orientation of the tracked object, tracking head movements with a device without a magnetometer represents a challenging task. To the best of our knowledge, because of the interference generated by the magnets of the speakers and in their cases, none of the earables in the market is equipped with a magnetometer. Indeed, like the eSense we are using in this work, the Apple AirPods[2], the Google Pixel Buds[3], and the Samsung Galaxy Buds[4] do not have a built-in magnetometer. *In this work, we focus on the evaluation of the eSense platform [7, 12] in tracking the head movements of a user concentrating on a specific spatial point.* To do so, we ran experiments with ten volunteers. We probed and stressed the robustness of the system by asking our volunteers to perform different activities, such as chewing and talking, while focusing on a series of targets placed at different spatial locations.

By tracking instantaneous head movements as a proxy to track visual attention, our study shows how a system, that relies only on accelerometer and gyroscope, can still provide

[1] Hearing disabilities are considered so when referring to hearing loss greater than 40dB in adult subjects and 30dB in children.

EarComp '19, September 10, 2019, London, UK

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[2] https://www.ifixit.com/Teardown/AirPods+2+Teardown/121471
[3] https://medium.com/@justlv/google-pixel-buds-teardown-396183cbbc18
[4] https://root-nation.com/audio-en/headphones-en/en-samsung-galaxy-buds-review/

useful insights on where a person is facing. Despite the fact that head movements are user dependent, we obtained estimations with an average error that ranges from 5.4 degrees for short movements done by silent subjects, to 18.7 degrees for longer movements carried out by subjects who are chewing. This paper lays the foundations of a line of work aiming to sense and characterize human attention through earables, wearables that are neither socially-awkward, nor cumbersome, unusable or unrealistic (e.g. combining an hearing-aid with a pair of eye-tracking glasses). Lastly, it sheds light on how a magnetometer would behave if placed in an earable.

2 RELATED WORK

Earables. Earables are a relatively new concept. Despite their huge potential, few research works currently use them. Most focus on health monitoring and sensing. For instance, LeBoeuf et al. [10] prototyped an optomechanical sensor to sense blood flow and estimate oxygen consumption during daily activities. The studies carried out by Bedri et al. [3] and by Amft et al. [2] combine IMUs and microphones to detect and classify eating activities. The former aimd to detect in-the-wild chewing activities, whereas the latter focused on the classification of four different types of food through the analysis of eating activities.

Head Movements Tracking. Inertial motion tracking is a known challenge and a well explored area. One of the most recent works in the field, and the state-of-the-art, is the study carried out by Shen et al. [13]. In their paper, the authors widely discuss the 3D Orientation problem and present MUSE, a magnetometer-centric sensor fusion algorithm for orientation tracking. Their results found that MUSE outperformed all the previous state-of-the-art orientation tracking approaches. Prior to their work, the other state-of-the-art techniques, as A^3 [15], were heavily relying on the gravity to determine the object orientation in the space, using the magnetometer data mainly to re-calibrate the system. However, as reported by Shen et al. [13], those previous works are mostly based on the following assumptions:

(1) slow linear motion, with accelerometer data that have gravity as average;
(2) slow rotational motion, with Gaussian errors that preserve the linearity of the system;
(3) motion with frequent, fairly long pauses, needed to reset the gravity estimation.

Yet, because of the huge freedom and unpredictability that characterizes human motion, these assumption rarely hold when tracking head movements. Our work partially follows the approach used by LaValle et al. [9]. However, in their work, the authors use a significantly different hardware (an Oculus headset), which is equipped with a magnetometer.

This paper assesses and evaluates the performance of eSense, an earble equipped solely with accelerometer and gyroscope, to track head movements. In addition, it presents a preliminary study on the effects of a magnetometer if placed in an earable. Given that there are no available earables equipped with this kind of sensor, we believe these preliminary observations represent and interesting input to both the research community and industry.

3 PLATFORM OVERVIEW

In this section we introduce the eSense platform and the challenges of performing head movement tracking on a device that can not rely on the data from a magnetometer.

eSense Platform

The eSense platform [7, 12] consists in a pair of true wireless earbuds which have been augmented with kinetic, audio and proximity sensing options. The left earbud has a 6-axis Inertial Measurement Unit (IMU) with accelerometer and gyroscope and a Bluetooth Low Energy (BLE) interface which is used to stream data and to send periodic beacons that can be used to detect proximity to nearby devices. Both earbuds are also equipped with microphones to record external sounds. The benefit of eSense, contrary to other commercial earbuds, is the access to the raw data from the onboard sensors and the complete flexibility in the configuration of their parameters. In addition to serve as a well established and socially acceptable device, for example to listen to music and take phone calls, eSense allows to gather real-time sensor data, opening the door to novel sensing applications involving the head.

Challenges of Inertial Tracking

The primary goal of this work is to understand the accuracy achievable by a device that solely relies on accelerometer and gyroscope to track user's head movements. The biggest challenges we had to face while investigating that, were related to the device itself. The small form factor, together with the pure "wireless experience" and the relatively short battery life , represent non-trivial constraints to deal with.

The presence of multiple magnets in the case compelled the hardware manufacturer to put a 6 degree of freedom (6 DoF) IMU, instead of a 9 DoF, more complete, sensor. Practically, it means the platform is bounded to the 3 DoF of the accelerometer (x_{acc}, y_{acc}, z_{acc}) and the 3 of the gyroscope (x_{gyro}, y_{gyro}, z_{gyro}), lacking the presence of a magnetometer.

Accelerometer and gyroscope provide relative movement estimates that drift over time. Without the magnetic north as a reference, we could not rely on the state-of-the-art calibration and re-calibration techniques [13]. Besides, when tracking the motion of an object (or of the head of a person), it is crucial to initialize the system correctly, with the right 3D orientation of the object itself. Unfortunately, once again,

the majority of the algorithms to solve the 3D Orientation Problem rely on the absolute direction reference provided by the magnetometer (combined with gravity and instantaneous gyroscope readings) [13, 15].

4 HEAD TRACKING METHODOLOGY

The data we collected from the eSense buds are accelerations (from the accelerometer) and rotational velocities (from the gyroscope). We had integrate the data to get an idea of where the person is paying its visual attention to.

Combining multiple IMUs. Tracking head motion from only one ear might not give precise enough results, especially considering the absence of external reference points, such as a magnetometer. As having more fine grained data points enhances the precision of the estimation, we combined the streams of inertial data coming from the IMU sensors in the two earbuds[5]. We leverage the assumption that the two different IMUs are recording, from different vantage points, the same rotation. Yet, combining the data correctly becomes crucial. We do so by concatenating and averaging the accelerometer and gyroscope data over a window of 200ms. Prior to that we resample and filter the readings from the IMU sensors. A further challenge comes from the orientation of the IMUs themselves. We obviate that by making our system independent from the human coordinate. To do that, we only consider the intensity of the rotation, rather the motion components along the 3 axis of the accelerometer and the 3 of the gyroscope. In fact, we only leverage the motion components to tell whether the rotation is positive (to the right) or negative (to the left).

Quaternions and Complementary Filter. Euler angles, better known by their components *yaw*, *pitch*, and *roll*, are the most common, and widely used, coordinate system to represents rotations. Despite their diffusion due to their ease of interpretability, they come with the problem know as Gimbal Lock [4]. To obviate the gimbal lock problem, it is common to switch to a better suited coordinate system: Quaternions [6, 9].

Integrated gyroscope measurements are subject to short-term drift, which may be more or less severe depending on the application. The situation worsen over time, as the error grows faster. To mitigate that, a common approach is to fuse the gyroscope readings with the accelerometer ones, as it is known to be more stable than the gyroscope in the short-time.

Our angular estimation is based on a complementary filter, which allows us to fuse gyroscope and accelerometer data, and follows the approach proposed by LaValle et al. [9]. We derive our estimation as follows:

$$q_{gyro} = \cos(\frac{\theta}{2}) + i\omega_x \sin(\frac{\theta}{2}) + j\omega_y \sin(\frac{\theta}{2}) + k\omega_z \sin(\frac{\theta}{2}) \quad (1)$$

where:
$\omega = (\omega_x, \omega_y, \omega_z) = (\frac{gyro_x}{||\omega||}, \frac{gyro_y}{||\omega||}, \frac{gyro_z}{||\omega||})$, and $\theta = ||\omega||dt$.

q_{gyro} is the quaternion that describes the instantaneous rotation of the head, based on the gyroscope data. We partially account for the gyroscope drift by adding the 3D orientation estimation and tilt correction of the head. To do that properly, we should rely on a combination of magnetometer and accelerometer data. Instead, since the eSense do not have a 9 DoF IMU, we could to only rely on the gravity as external reference to perform a rough orientation estimation and tilt correction.

$$\begin{gathered} q_{acc\_body} = 0 + i\ acc_x + j\ acc_y + k\ acc_z \\ q_{acc\_world} = pos[t] * q_{acc\_body} * pos[t]^{-1} \\ q_{cf} = \cos(\frac{\phi}{2}) + i\frac{n_x}{||n||}\sin(\frac{\phi}{2}) + j\frac{n_y}{||n||}\sin(\frac{\phi}{2}) + k\frac{n_z}{||n||}\sin(\frac{\phi}{2}) \end{gathered} \quad (2)$$

where q_{cf} is the implementation of the complementary filter in the space of quaternions, and where:

$$\begin{aligned} \phi &= (1-\alpha)\arccos(\frac{q_{acc\_world_y}}{||q_{acc\_world}||}) \\ v &= (\frac{q_{acc\_world_x}}{||q_{acc\_world}||}, \frac{q_{acc\_world_y}}{||q_{acc\_world}||}, \frac{q_{acc\_world_z}}{||q_{acc\_world}||}) \\ n &= v * (0,1,0) \end{aligned}$$

We can now estimate the final rotation by doing:

$$final\_position = q_{cf} * pos[t] \quad (3)$$

Notice that for each earbud, we account for the factory offset of both accelerometer and gyroscope by using the techniques described by Kok et al. in their work [8]. In addition, because of the absence of the magnetometer, we only focus on relative rotations (*delta motions*).

5 USER STUDY

In this section we describe the methodology we followed to investigate head motion tracking through earables. We detail how we collected the data and the results obtained tracking the head movements of our volunteers. Ethical approval was obtained to conduct the user study.

Data Collection Methodology

We recruited 10 volunteers to join our data collection campaign. Each individual was wearing two earbuds (both eSense left bud) connected via BLE (Bluetooth Low Energy) to an Android application running on a smartphone[6] provided by us. The eSense buds collected, and streamed to the smartphone, at 100 Hz. For the sake of reproducibility we report the configuration of the two earbuds. Notice the two buds have the same configuration:

- $AccelerometerRange = \pm 2g$
- $GyroscopeRange = \pm 500 degrees/second$

[5] since only the left eSense buds are equipped with the IMU sensors, we had to use two left buds.

[6] Google Pixel 2, https://en.wikipedia.org/wiki/Pixel_2

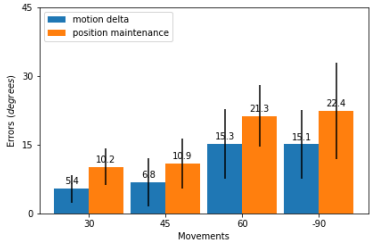


Fig. 1: Mean error and standard deviation of the head movements estimation of 10 silent volunteers.

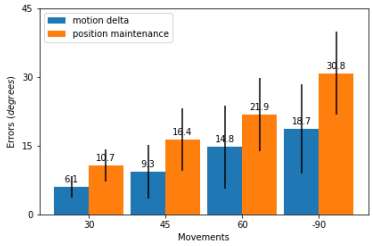


Fig. 2: Impact of chewing activity on the mean error and standard deviation of the head movements estimation of 10 volunteers.

- $AccelerometerLowPassFilter = 5Hz$
- $GyroscopeLowPassFilter = 5Hz$

The experimental set up consisted of 4 targets (red cross) attached to the walls in an empty room. The angle between the position of the volunteer and the targets was known and represented ground truth. The targets were placed respectively at 30, 45, 60, and -90 degrees. We chose 30 degrees as the smallest angle we investigate assuming that for smaller angles people would mostly move their eyes, barely moving their heads. With -90 degrees, we wanted to show how our system could capture both clockwise and counterclockwise movements. We asked our volunteers to perform the following actions:

- Standing in silence and looking at different targets, and keep facing them, according to the instructions of the investigators;
- Standing, chewing a piece of chewing gum, and looking at different targets, and keep facing them, according to the instructions of the investigators;
- Standing, conversing with one of the researchers, and looking at different targets, and keep facing them, according to the instructions of the investigators.

Baseline: Silent Subject

We now present the results of what we consider our baseline. In this experiment, the volunteers were standing in silence, looking at the different targets according to a set of instructions provided by us. For each target, the volunteers started

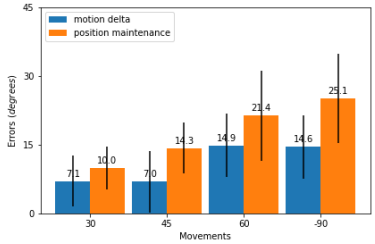


Fig. 3: Impact of speech on the mean error and standard deviation of the head movements estimation of 10 volunteers.

facing an initial reference, placed at 0 degrees. They then rotated their heads towards the given target (*delta motion*). Once there, we asked them to keep their head turned towards the target for about 5 seconds (*position maintenance*). We estimated the movements done by the volunteers processing the readings of accelerometer and gyroscope according to what described in Section 4. Figure 1 respectively reports the mean errors of the motion delta estimation and the position maintenance estimation when the users were rotating their heads clockwise 30, 40, and 60 degrees, and counterclockwise to -90 degrees. From the chart, we can immediately appreciate how the position maintenance errors are greater than the motion delta ones. This is due to the inertial sensors' drift that heavily affects the integration. Because of our long term application, we are interested in instantaneous movements (deltas) we therefore mostly care about motion delta errors. Another interesting observation is how the errors grow for longer movements (i.e. greater angles), indicating a higher precision of the system for short movements (i.e. small angles). Moreover, the high standard deviation that characterizes the mean errors denotes a strong user dependency of the motion estimation. In fact, for some volunteers, we even registered sub-degree motion delta accuracy in some movements.

Impact of Chewing Activity

Once we evaluated our system on the simplest case, we started testing the robustness of our motion estimation. We gave chewing-gum to our volunteers, and asked them to repeat the same sequence of movements. The chewing activity of the volunteers generated spurious vibrations that were inevitably picked up by the inertial sensors in the earbuds. To make our system more robust to this kind of noise, we tuned the parameter α of our complementary filter, aiming for the best performance in all the three types of experiments. Figure 2 depicts the mean errors of the estimation. If compared with our baseline, as expected, the errors are slightly higher. As we observed in the previous case, because of the drift, the motion delta mean errors are smaller than the position maintenance ones.

Impact of Speech

Lastly, we assessed how speech affects our estimation. To do so, we asked our volunteers to talk. As for the chewing experiment, speaking generates unwanted vibrations and micro-movements that are captured by the sensors, and we are interested to study the robustness of our system. Figure 3 shows how our system behaves with talking subjects. As before, the mean errors of the estimation in this third case are sightly higher than the ones for silent users. As in both the previous cases, the motion delta accuracy is higher and decreases for longer movements. Comparing Figure 2 and Figure 3 we can observe that chewing activity seems to have a comparable impact on our motion estimation as speech. The high standard deviation of the mean errors further remarks the findings we got from the previous experiments.

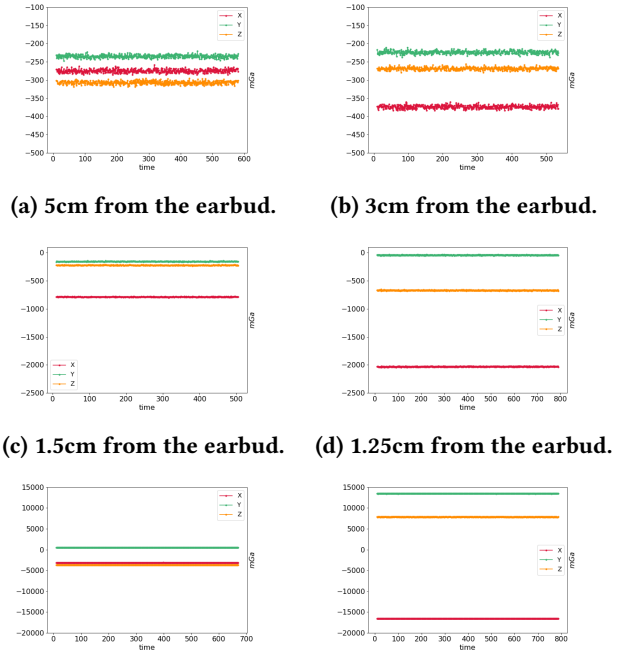
Summary. Evaluating the performance of the eSense platform, our work investigates the potential of ears as a vantage point to sense visual attention through head movements. We achieve estimations with an average error that ranges from 5.4 degrees for short movements in the least challenging situation, to 18.7 degrees for longer movements, in noisier circumstances. In the reminder of this paper, we try to further improve the accuracy of our system, investigating whether it would be feasible add a magnetometer, thus gaining an external absolute reference point. This would allow us to track absolute and incremental movements and not only relative motion.

6 WHY IS THE MAGNETOMETER MISSING?

In order to asses if we could use state-of-the-art motion tracking approaches to further improve the precision of our motion estimation, we studied how the magnets used to hold the earbuds into the case and the magnet in the speaker affect the readings of a magnetometer. We did that by placing a STEVAL-STLCS01V1 sensor tile[7] at different distances from one earbud, and we plotted the data captured by the magnetometer in Figure 4.

To start off, we put down the magnetometer at a distance of 5 cm from the earbud (Figure 4a). We proceeded keeping the earbud still in the initial position, while moving the sensor tile closer to the bud. Figure 4b shows the magnetometer readings when 3 cm apart. We can immediately appreciate how the magnets start affecting the data, introducing an offset. It is worth noticing how the offset does not fluctuate when the magnetometer is fixed. As expected, the readings change at different distances from the earbud. The closer we get to the earbud, the higher the influence of the magnets is. We further moved the STEVAL-STLCS01V1 at a distance of 1.50 cm, 1.25 cm, and 1 cm, as respectively depicted in Figure 4c, Figure 4d, and Figure 4e. Here, we can clearly notice

[7] https://www.st.com/en/evaluation-tools/steval-stlcs01v1.html



(a) 5cm from the earbud. **(b) 3cm from the earbud.**

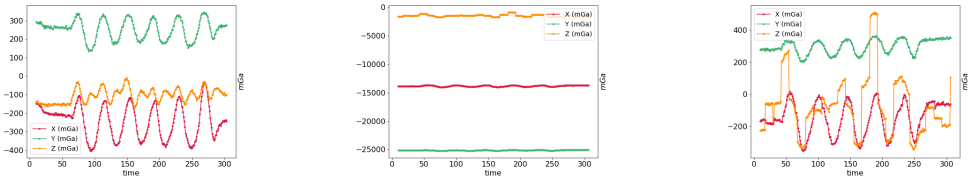
(c) 1.5cm from the earbud. **(d) 1.25cm from the earbud.**

(e) 1cm from the earbud. **(f) Inside the earbud case.**

Fig. 4: Magnetometer readings from a STEVAL-STLCS01V1 device at different distances from one eSense earbud. Notice how the scale of the plots changes from top to bottom.

how the magnetic field generated by the magnets in the earbud overtakes the Earth's one, flattening all the readings. Eventually, we put the sensor tile inside the case (Figure 4f). The readings skyrockets, as the earbud's magnetic field adds a significant offset to the Earth's.

Because of the constant offset at different locations, we decided to delve deeper into the behaviour of the magnetometer, collecting more data samples while moving the device. We repeated the same set of movements twice. In the beginning, we moved the STEVAL-STLCS01V1 alone, without any direct external interference caused by either the earbud or by the vicinity of a metallic source (Figure 5a). In the following experiment, we recorded the data, performing the very same movements, but placing the STEVAL-STLCS01V1 inside the case of the earbud, swapping it with the eSense's existing PCB (printed circuit board) (Figure 5b). While in the first case, where there was not interference, the magnetometer was able to record the motion events, from Figure 5b we are unable to observe any motion-related data. However, the constant trend of the offset let us apply the calibration technique know as Hard Iron Distortion. As a result, we managed to recover most of the motion related information, especially along x and y (Figure 5c). These preliminary results provide an initial indication about the possibility of integrating a magnetometer even with the presence of strong magnets in the sensor's vicinity. However, experiments with different



(a) No interference. (b) Inside the case. (c) Distortion correction.

Fig. 5: We run two experiments where a volunteer was asked to shake his head while having the a STEVAL-STLCS01V1 close to the ear. (a) We first collected data without magnetic interference. (b) We then asked our volunteer to repeated the same movements with the STEVAL-STLCS01V1 placed inside the eSense case. (c) This way we could observe how correcting the offset introduced by the magnets, the magnetometer placed in the earbud is still able to record motion data.

conditions (e.g. when the buds are playing music) and a detailed analysis of the resulting data are needed to confirm our insights. We leave this for future work.

7 FUTURE WORK

Head movement tracking represents only a first step towards the characterization and sensing of human attention. The behavioural cues we want to sense are not only limited to head movement. After studying how the magnetometer would work if placed in the case, and after evaluating the eSense platform to track head motion, this paper motivates a second revision of the hardware. This should include a wide variety of sensors, such as:

- 9 DoF IMU;
- EOG support for gaze tracking;
- electroencephalography (EEG) sensors.

Such a device, could represent a step towards a platform to sense attention and act as an advanced hearing-aid. We acknowledge that the suggested improvements will pose significant technical challenges, such as battery-life and form factor, which we will gradually explore and tackle in future work.

8 CONCLUSION

In this work, we evaluated eSense as an earable device to perform in-ear head motion tracking. Our technique combines multiple streams of data, and, despite the absence of a magnetometer in the inertial sensor equipped in eSense, achieves results precise up to a few degrees. Although the accuracy of our estimation decreases for longer movements, it performs well also in more realistic situations (e.g. with the subjects speaking or chewing). Besides, our preliminary study on the magnetometer represents an interesting input to take into account in the development of future earables.

ACKNOWLEDGMENTS

This work is supported by Nokia Bell Labs through their donation for the Centre of Mobile, Wearable Systems and Augmented Intelligence.

REFERENCES

[1] World Health Organization. Deafness and hearing loss. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss, 2019.

[2] O. Amft, M. Stäger, P. Lukowicz, and G. Tröster. Analysis of chewing sounds for dietary monitoring. In *Proceedings of the 7th International Conference on Ubiquitous Computing*, UbiComp'05, pages 56–72, Berlin, Heidelberg, 2005. Springer-Verlag.

[3] A. Bedri, R. Li, M. Haynes, R. P. Kosaraju, I. Grover, T. Prioleau, M. Y. Beh, M. Goel, T. Starner, and G. Abowd. Earbit: Using wearable sensors to detect eating episodes in unconstrained environments. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3):37:1–37:20, Sept. 2017.

[4] J. Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix*, 58(15-16):1–35, 2006.

[5] A. Favre-Félix, C. Graversen, R. K. Hietkamp, T. Dau, and T. Lunner. Improving speech intelligibility by hearing aid eye-gaze steering: Conditions with head fixated in a multitalker environment. *Trends in Hearing*, 22:2331216518814388, 2018.

[6] W. R. Hamilton. *Lectures on Quaternions: Containing a Systematic Statement of a New Mathematical Method.* 1853.

[7] F. Kawsar, C. Min, A. Mathur, and A. Montanari. Earables for personal-scale behavior analytics. *IEEE Pervasive Computing*, 17(3):83–89, Jul 2018.

[8] M. Kok, J. D. Hol, and T. B. Schön. Using inertial sensors for position and orientation estimation. *arXiv preprint arXiv:1704.06053*, 2017.

[9] S. M. LaValle, A. Yershova, M. Katsev, and M. Antonov. Head tracking for the oculus rift. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 187–194, May 2014.

[10] S. F. LeBoeuf, M. E. Aumer, W. E. Kraus, J. L. Johnson, and B. D. Duscha. Earbud-based sensor for the assessment of energy expenditure, hr, and vo2max. *Medicine and science in sports and exercise*, 46 5:1046–52, 2014.

[11] J. H. Mcdermott. The cocktail party problem. *Curr. Biol.*, 19(22):1024–1027, 2009.

[12] C. Min, A. Mathur, and F. Kawsar. Exploring audio and kinetic sensing on earable devices. In *Proceedings of the 4th ACM Workshop on Wearable Systems and Applications*, WearSys '18, pages 5–10, New York, NY, USA, 2018. ACM.

[13] S. Shen, M. Gowda, and R. Roy Choudhury. Closing the gaps in inertial motion tracking. *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking - MobiCom '18*, 2018.

[14] B. G. Shinn-Cunningham and V. Best. Selective attention in normal and impaired hearing. *Trends in Amplification*, 12(4):283–299, 2008.

[15] P. Zhou, M. Li, and G. Shen. Use it free: Instantly knowing your phone attitude. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 605–616, New York, USA, 2014. ACM.

Can Earables Support Effective User Engagement during Weight-Based Gym Exercises?

Meera Radhakrishnan
Singapore Management University
meeralakshm.2014@phdis.smu.edu.sg

Archan Misra
Singapore Management University
archanm@smu.edu.sg

ABSTRACT

We explore the use of personal 'earable' devices (widely used by gym-goers) in providing personalized, quantified insights and feedback to users performing gym exercises. As in-ear sensing by itself is often too weak to pick up exercise-driven motion dynamics, we propose a novel, low-cost system that can monitor multiple *concurrent* users by fusing data from (a) wireless earphones, equipped with inertial and physiological sensors and (b) inertial sensors attached to exercise equipment. We share preliminary findings from a small-scale study to demonstrate the promise of this approach, as well as identify open challenges.

CCS CONCEPTS

• **Information systems** → **Mobile information processing systems**; • **Applied computing** → *Health care information systems*.

KEYWORDS

Gym Exercises, Personalized Coaching, Earable, IoT

ACM Reference Format:
Meera Radhakrishnan and Archan Misra. 2019. Can Earables Support Effective User Engagement during Weight-Based Gym Exercises?. In *Proceedings of ACM Conference (UbiComp/ ISWC'19 Adjunct).* ACM, London, UK, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

While there has been a rapid increase in the market for fitness devices and apps, relatively few solutions offer quantified and personalized feedback on an individual's overall exercise-related activities [9]. Existing technologies for fine-grained, individualized exercise tracking typically utilize video-based sensing [16], WiFi CSI information [7], or on-body wearable devices [4, 11]. However, such solutions continue to face challenges in real-world adoption. For example, video-based sensing generates significant privacy concerns, WiFi solutions suffer from poor accuracy in the presence of multiple individuals (e.g., at a gym) and individuals are reluctant to adopt custom wearable devices, *unless the wearable device is already a part of an individual's lifestyle*.

Motivated by these observations, we investigate the possibility of tapping on ear-worn (*'earable'*) devices (such as in-ear earphones) as a possible means of *capturing* a user's exercise related activities. Earables offer a compelling and attractive *mass-market* wearable platform ([14] reported a global sale of 368 million headphones and headsets in 2018). Moreover, they are also commonly used during gym activities (e.g. for listening to music while working out). They also offer the advantage of supporting *real-time*, personalized *audio-based* feedback (often preferred to alternative text-based feedback [10])–for example, to rectify incorrect exercising behavior or to motivate continuation of desirable activities.

Key Challenge: The big drawback of earables, of course, is their unfavorable on-body placement: it is indeed questionable whether ear-based inertial signals can provide *any* discriminative information about exercise motion, especially when such motion is primarily restricted to upper or lower limbs. Research on earable-based activity recognition has been confined to inferring (a) characteristics of eating or drinking [3], both of which obviously manifest in head motion, and at a stretch, (b) high-level locomotive activities [12], which also involve overall body displacement. To our knowledge, no prior work has tackled the problem of fine-grained monitoring of gym exercises using earables.

This paper introduces a novel, low-cost solution for earable-based, individual-specific *fine-grained* monitoring of gym exercises in real world scenarios, *where multiple individuals are exercising concurrently*. Our key insight is that earable-based sensing, in isolation, is too noisy and weak to directly offer accurate recognition of gym activities. To overcome this limitation, we propose a hybrid architecture (to be elaborated in Figure 1), consisting of:

UbiComp/ ISWC'19 Adjunct, September 9-13, 2019, London, UK

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

- Individuals wearing wireless earphones embedded with sensors (e.g., inertial sensors, heart rate sensor) that capture their activity and physiological context
- Individual gym equipment (e.g., dumbbells, weight machine) attached with cheap IoT sensors that capture the motion dynamics of each equipment.

Given this architecture, the problem then morphs to (a) first establishing an *association* between an individual's earable device and the corresponding gym equipment, and (b) then using this pair of (earable, equipment) sensor data to infer fine-grained aspects of the exercise being performed. While not part of this paper, our overall vision also involves the generation of personalized real-time audio-based feedback (acting as a *"virtual personalized exercise coach"*), to the exercising individual, based on such fine-grained insights.

Key Contributions: Using a very preliminary feasibility study (multiple exercise sessions conducted with two users concurrently performing identical or distinct exercises using dumbbells or weight-stack machines), we demonstrate that:

- Even though exercise-related signals are often very muted on an earable, it is indeed possible to identify related (earable, equipment) pairs from the combined inertial sensor data, via the application of sophisticated time-frequency domain statistical correlation techniques. Our resulting analysis reveals high correlation (>0.71) between the *earable* and *dumbbell* signals corresponding to the same user, and 83% accuracy in pairing the devices used by multiple *concurrent* users.
- By fusing sensor data from both the earable and equipment-embedded inertial sensors, we can obtain fine-grained insights into an individual's exercise patterns (e.g., exercise type). In particular, we show that such fusion can determine the exercise performed (among 8 candidate exercises) with an accuracy of 92%, higher than that can be achieved from either modality in isolation.

Overall, our work provides early evidence of the promise of earable devices as a platform for capturing fine-grained context of individuals exercising in a gym.

2 SYSTEM ARCHITECTURE

A future smart gym application should have the following capabilities: (i) distinguish between multiple people exercising simultaneously in the gym, (ii) unobtrusively monitor exercises performed by each individual and obtain deeper insights on various facets of exercising, (iii) provide personalized feedback to the individuals to improve the exercise effectiveness and prevent injuries.

For realization of such a smart gym application, we assume that individuals exercising in the gym are using *earables* and the exercise equipment/machine is attached with cheap IoT sensor devices. The earables are equipped with a microphone,

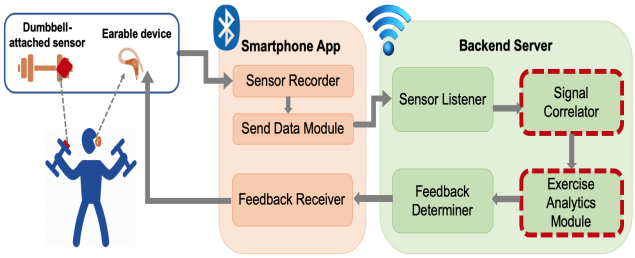


Figure 1: System Architecture

inertial sensors (accelerometer, gyroscope), bio-sensors (heart rate, body temperature) and are paired to a smartphone. The IoT device attached to the exercise equipment (e.g., dumbbells, barbells, weight machines) have embedded accelerometer, gyroscope and magnetometer sensors. A custom built smartphone application has a *Sensor Recorder* process that records the sensor data from both the devices and a *Send Data* module that periodically transmits the sensor data to a backend server over the WiFi network. This App also has a *Feedback Receiver* that receives audio inputs/feedback from the server and relays it to the earables.

The backend server performs all the smart gym analytics. In the backend, there is a *Sensor Listener* module for obtaining sensor data from both the earable and the equipment-sensor. Once the sensor data is obtained, the *Signal Correlator* module checks for the correlation between the earable sensor stream and equipment sensor stream to determine who is working out with which exercise equipment. The correlated sensor data pairs are then fed to the *Exercise Analytics* module, which identifies the type of the exercise performed and determines more fine-grained aspects such as the exercise intensity, correctness, heart rate variation for different exercises. Then, the *Feedback Determiner* module utilizes these analytics to determine the appropriate timing and the audio feedback to be sent to the earable device.

Figure 1 illustrates the architecture of the system with the sensor devices, server components and flow of the analytics pipeline. In this work, we mainly focus on the two components outlined in red-dotted lines. *Note:* For a clear representation, the figure depicts only a single-user scenario. In a practical setting, there will be multiple people exercising and thus multiple streams of both dumbbell and earable sensor will be streamed simultaneously to the backend sever.

3 EARABLE-BASED INERTIAL SENSING FOR EXERCISE ACTIVITY RECOGNITION

In this work we focus on answering the following **key research questions:**

- Does the accelerometer on the ear-worn sensor device show any discernible pattern for the common weight training exercises performed by individuals in a gym?

- Can we correlate the sensor data from the ear-worn device and the dumbbell-attached device to distinguish between individuals and identify the exercise performed by each person?

We next describe our study procedure, data collected and the overall approach of analyzing sensor data and deriving various insights on the exercises performed concurrently by multiple individuals in the gym.

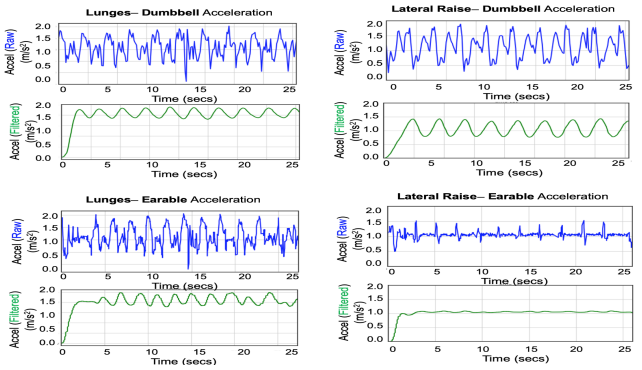
Study Procedure and Data Collection

To study the feasibility of our proposed vision, we conducted multiple studies in our campus gym. For obtaining sensor data, we used the following devices: (i) eSense Earable device[1], which the subjects wore on their left ear, (ii) Cosinuss One[2] earphone, worn by subjects on their right ear and (iii) a multi-sensor device (DA14583 IoT Sensor[3]) to attach to the exercise equipment (e.g., dumbbells, exercise machines). For the eSense earable, we used only the left-side earbud which has the capability to stream inertial sensor (accelerometer and gyroscope) data as well as receive audio inputs. The Cosinuss One device has in-built sensors to record heart rate and body temperature. These devices are paired with a smartphone and we developed an android application that simultaneously connects to these devices over Bluetooth Low Energy (BLE) and records sensor data and also records ground truth labels such as exercise performed, set count and amount of weight lifted. *Note:* In this work, all the chosen free-weights exercises are performed with two arms moving at the same time and we attach a sensor device only to a single dumbbell (left-hand).

For the study, we obtained data from 2 subjects who performed 8 different weight training exercises across 3 days. In each session, the subjects performed 3 sets of 10 repetitions of each exercise. The exercises involved 6 dumbbell exercises: (i) Biceps Curls, (ii) Triceps Pushdown, (iii) Lateral Raise, (iv) Side Bend, (v) Squats, (vi) Lunges and 2 exercises on weight stack machines: (vii) Standing Cable Lifts (for Abs) and (viii) Bent Over Side Lateral (for Shoulders). Out of the three sets of each exercise in a session, both subjects simultaneously performed the same exercise for 2 sets and for the last set, they alternated between different exercises. Overall, we collected 144 sets (of 10 reps each) of exercise data.

Sensor Data Analysis and Insights

We first inspect the accelerometer data recorded from the eSense left earbud and the dumbbell sensor. As expected, the dumbbell accelerometer showed clear and varying patterns for most of the exercises. For the earable, as any 'exercise-related' perturbations, if they exist, will be minor and may



(a) Raw (blue) and filtered (green) signals from Dumbbell (top) and Earable (bottom) for *Lunges* exercise

(b) Raw (blue) and filtered (green) signals from Dumbbell (top) and Earable (bottom) for *Lateral Raise* exercise

Figure 2: Accelerometer Sensor Patterns from Dumbbell & Earable for (a) Lunges and (b) Lateral Raise exercises

get swamped by various other macro-movements, we first pre-process and filter the sensor data. For this, we analyze the typical 'exercising frequency' of various exercises from the dumbbell sensor pattern. We observe that on an average the time taken to complete one repetition of a dumbbell/machine exercise is about $2-2.5$ seconds. As such, we use a fourth order Butterworth band pass filter with a lower cut off frequency of 0.4 Hz and a higher cut off frequency of 4 Hz to filter both streams of sensor data.

Figure 2 shows sample plots of the magnitude of the raw and filtered sensor signals for *Lunges* and *Laterl Raise* exercises. We find that exercises which involve larger body movements (for e.g., *lunges*, *squats*, *abs* exercise on machine) exhibit clear patterns in the earable signal for each exercise repetition. However, for certain upper-arm exercises (such as *biceps curls*, *lateral raise*), variations are not clearly evident in the time-domain earable signal. This makes the problem both promising and challenging and requiring further analysis of both time and frequency domain of the signals.

Identifying the Correct User-Dumbbell Pairs

In our targeted gym scenario, multiple users would perform exercises simultaneously and the *smart gym* application should monitor exercise and provide personalized feedback to each individual. As such at the server side, we would receive multiple streams of both *earable* and *dumbbell* signals and therefore, our primary goal is to identify the correct pairs of $\{earable - dumbbell\}$ sensor streams to determine who is exercising with which dumbbell.

For this purpose, we propose to first obtain *Continuous Wavelet Transform* (CWT) of the signals and then perform correlation analysis in the frequency domain. We choose to

[1] eSense– http://www.esense.io/
[2] Cosinuss One– https://www.cosinuss.com/products/one/
[3] DA14583 IoT Sensor – (https://www.dialog-semiconductor.com/iotsensor)

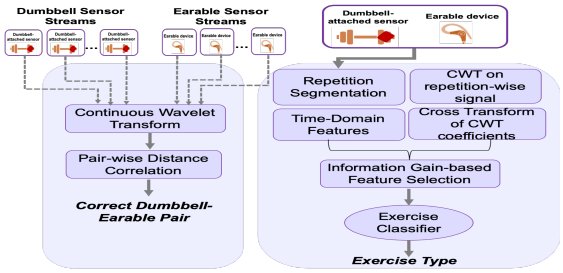


Figure 3: Steps involved in identifying the Correct *{Dumbbell-Earable}* Pair and Exercise Type

use wavelet decomposition instead of other frequency domain techniques such as Fourier Transform or Power Spectral Density because of its ability to obtain both temporal and frequency resolution of the analyzed signal.

The CWT coefficients are computed at different scales for each of the filtered earable and dumbbell sensor streams. We use the *Symlet* wavelet ('sym4') and vary the scales from 1 to 100. After performing CWT, we obtain a wavelet coefficient matrix from both sensor streams of all exercising individuals. Next, we compute *distance correlation* between all the possible pairs of the coefficient matrices. The distance correlation is a measure of dependence between random vectors and is obtained by dividing the distance covariance of two matrices by the product of their distance standard deviations [15].

To identify the correct *User-Dumbbell* pair, we further train a binary-class classifier with all combinations of CWT matrices from the dumbbell and earable signal as input. i.e., Given a dumbbell signal X and an earable signal Y, does the classifier think that (X, Y) is from the same pair or not?

Identifying Exercise Type

Once the correct dumbbell-earable pairs are determined, the next step is to identify the exercise performed by each individual. We use a supervised ML classifier trained on features extracted from both the dumbbell and earable signals (Figure 3 depicts the various steps involved). Using peak and valley detection on the filtered dumbbell sensor signal for an exercise set, we first perform repetition identification and segmentation. We assume one repetition to be the time segment between two consecutive valleys. We then compute the continuous wavelet transform of each repetition segment of both dumbbell and earable signal. Then the cross-transform/convolution of the two wavelet coefficient matrices is computed. We also compute other statistical time-domain features on both sensor streams. Using *InformationGain*-based feature selection, we found that CWT coefficients from scale 70 to 100 and 1 to 10 are the most informative. So, we ignored the features corresponding to CWT scales 11 to 69 from the feature set. The exercise classification is then performed on the new feature set with a Random Forest (RF) classifier. We also compare the performance with other machine learning classifiers.

4 PRELIMINARY RESULTS

In this section, we describe our early results in distinguishing between multiple exercising individuals and identifying the exercise they perform, based on the data we collected from our campus gym.

Performance of Identifying the Correct Pairs

As our system is intended for multi-user gym environments, our primary goal is to distinguish between the different people exercising in order to perform personalized exercise monitoring. We achieve this by correlating sensor signals from the dumbbell and earable in the frequency domain. As discussed earlier in Section 3, we first perform continuous wavelet decomposition of both the sensor streams. Figure 4 plots the scalogram (which is the absolute value of the CWT coefficients of a signal, plotted as a function of time and scale) of one set of *Side Bend* exercise. From the figure, we can see that the individual exercise repetitions have their energy concentrated between scales 60 to 100. We observe similar trends for other exercises as well.

As we collected data with two people exercising simultaneously, for each exercise set, we first obtain four CWT coefficient matrices (for the dumbbell and earable data from each user) and then compute the distance correlations between them. We observe that, on an average, the correct pair of signals (i.e., from same user's dumbbell and earable) have high correlation value over **0.71**.

To automatically classify the correct and incorrect pairs, we train a Random Forest classifier and evaluate the performance using 10-fold cross validation. We obtain an accuracy of **83%** in identifying the correct pair. Upon analyzing the incorrectly classified instances, we gather two insights: (i) several of the mis-classified instances belongs to the sets where both subjects were concurrently performing the "same" exercise and (ii) sets of *biceps curls* and *lateral raise* exercises (which involve limited head motion) have comparatively more number of mis-classifications.

Performance of Identifying Exercise Performed

We next evaluate the accuracy of classifying the 8 exercises (dumbbell and machine exercises) from a total of 144 sets of exercise data collected. We have a balanced set of data as we collected equal number of sets for each exercise. We then perform 10-fold cross-validation and report the average performance metrics in Table 1, for a number of machine learning algorithms in Weka.

We observe that the highest performance is obtained with a Random Forest classifier, with an average accuracy of **92.9%**, precision of 0.93 and recall of 0.929 in classifying the 8 exercises. On inspecting the confusion matrix (see Figure 5), we found that the classification errors occurred primarily for

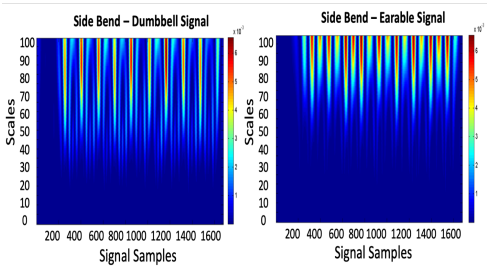


Figure 4: Scalogram of Dumbbell (*left*) and Earable (*right*) signal for *Side-Bend*

	Accuracy	Precision	Recall
Naive Bayes	79.23%	0.792	0.79
Random Forest	**92.94%**	**0.93**	**0.929**
Decision Tree	83.98%	0.84	0.84
SVM	71.04%	0.71	0.71
Logistic	72.98%	0.73	0.73

Table 1: Exercise Classification accuracy using different machine learning algorithms

Predicted Label	Actual Label							
	Squats	Lunges	Side-Bend	Biceps	Triceps	Lat-Raise	Shoulders	Abs-Lift
Squats	97.19%	0.29%	1.23%	1.40%	0.35%	0.26%	0.34%	0.23%
Lunges	1.50%	97.16	2.37%	0.45%	1.88%	0.88%	0.52%	0.11%
Side-Bend	0.13%	0.83%	95.28%	0.84%	0.39%	0.85%	0.46%	0.06%
Biceps	0.28%	0.40%	0.48%	89.40%	5.73%	1.98%	1.12%	0.62%
Triceps	0.86%	0.32%	0.43%	4.66%	88.11%	1.88%	1.06%	1.02%
Lat-Raise	0.04	0.68%	0.18%	1.69%	1.64%	91.59%	0.33%	0.64%
Shoulders	0.00%	0.26%	0.02%	0.73%	1.35%	1.74%	90.85%	3.91%
Abs-Lift	0.00%	0.06%	0.01%	0.83%	0.55%	0.82%	5.32%	93.41%

Figure 5: Confusion Matrix of Exercise Classification with RF Classifier

the following exercises: *triceps pushdown* & *biceps curls* (dumbbells) and *bent over side lateral* (machine), which has comparatively lesser head movements involved.

We next investigate the performance of exercise classification when only *either* of the dumbbell or earable sensor data is used to train the model. We obtain an average accuracy of *85.20%* and *54.58%* by considering only dumbbell or only earable data, respectively. Although the average accuracy obtained with earable is quite low, we observe that classification of certain exercises (e.g., lunges, side bend) achieves a higher precision of ≈ 0.78 with earables. This shows that combining both dumbbell and earable data helps to increase the performance of classifying exercises.

5 DISCUSSION

Our initial results are promising, but admittedly based on a small scale study conducted with only two users. Further extensive studies with a larger population and more number of people exercising simultaneously needs to be conducted to validate our approach. There are several other aspects and open questions that we are actively pursuing to make this vision an eventual reality.

Real-time audio feedback: Providing personalized feedback on the exercise progress and correctness could help improve exercise effectiveness as well as retain motivation to continue exercising. Prior studies [10] have reported that "auditory feedback" is ranked top among feedback features based on a review of physical activity apps. Based on real-time sensing and analysis of the multi-modal sensor data, we intend to provide incremental feedback in the form of short audio instructions or "beep" sounds, based on the performance and progress of users. The system could also provide positive, motivating feedback after completing each exercise set and at the end of the gym workout for the day. Motivated by prior work [12], we shall also investigate if we can use music to regulate the 'exercise tempo' of users.

Integrating physiological sensor data: Besides inertial sensor data, additional physiological signals (e.g., heart rate or breathing rate) from earables may enable more sophisticated monitoring or intervention strategies. For example, the physiological data could reveal the user-perceived intensity of the current exercises and enable the delivery of appropriate corrective feedback–e.g., alerting the user to slow down if the heart rate exceeds the maximum active heart rate. More interestingly, such physiological signals may provide additional temporal markers for better matching of {earable-dumbbell} pairs, especially for exercises with imperceptible head motion–e.g., if the inhalation/exhalation times match with the exercise repetition dynamics.

Robust and generalized pairing: We will need to extend our 'matching' technique to scenarios with a larger number of concurrent users. Moreover, we would have to incorporate practical situations where all exercising individuals may not be wearing earable devices–in such cases, we would obtain M earable and N dumbbell sensor streams ($M < N$). To tackle such scenarios, we plan to first assign 'confidence scores' to different {earable-dumbbell} pairs, and then apply *inexact bipartite matching* techniques to improve the association of users to specific exercise equipment.

Extending to other exercise types and scenarios: In this work, we focus only on weight training exercises (both free-weights and machine weights). However, we believe that the ear-worn sensing platform can be used to monitor other types of gym exercise (e.g., cardio, body-weight exercises) and other outdoor exercises or sports. Additionally, the proposed approach of real-time sensing of activities and bio-signals using in-ear sensors can also be extended to other lifestyle activities such as monitoring cognitive state and well-being of people in office environments.

6 RELATED WORK

We highlight recent work on monitoring gym exercises, as well as work that is most closely related to our vision of using ear-worn sensors for activity recognition.

Pervasive Monitoring of Gym Exercises: In the recent years, several commercial mobile applications (e.g., Trackmyfitness[4], JEFIT[5] and wearable devices (e.g., Apple Watch, Nike Fuelband) have spawned in the fitness space with the goal to digitally track and encourage physical activity among individuals. However, a review of such physical activity apps

[4]Trackmyfitness– http://trackmy.fit
[5]JEFIT– https://www.jefit.com/

found that only 2% provided evidence-based guidelines for gym exercises training and people find it not helpful [9].

Existing pervasive technologies for providing quantified insights into an individual's gym activity rely primarily on on-body wearable devices (e.g., [4, 11]) and video-based sensing [8, 16]. However, each of these approaches have different drawbacks such as usability concerns with wearables and the reluctance to wear such additional devices while exercising in gym, the overly intrusive nature and privacy concerns associated with videos. Guo et al. [7] uses Wi-Fi CSI information to analyze workouts within a home/work environment. However, WiFi-based systems may not work in a multi-user gym environment and in non line-of-sight scenarios. We believe that earables present a compelling alternative because of its form-factor as well as the wide use of earphones by exercisers during gym activity. The FEMO [5] system and the recently proposed JARVIS system [13] rely on the idea of attaching sensors to exercise equipment (dumbbell or weight machine) to track various aspects of specific class of gym exercises. In our work, we propose a hybrid approach of combining sensor data from earables as well as equipment attached sensor device to obtain accurate and fine-grained tracking of the exercises performed in a gym.

Ear-worn Sensing for Activity Recognition: While prior works have explored the use of microphones in ear-worn devices to capture chewing sounds [1] and eating episodes [3], not many has explored the use of inertial sensors on ear-worn devices for complex activity recognition. Atallah et al. [2] proposed using an ear-worn accelerometer for gait monitoring while exercising on a treadmill. Nirjon et al. [12] proposed the 'MusicalHeart' system which uses a sensor-equipped ear-worn device that monitors heart rate and provides music recommendation based on user's activity levels. Gil et al. [6] developed a prototype of an ear-worn device that can measure cardiovascular and sweat parameters during physical exercise.

7 CONCLUDING REMARKS

In this work, we have introduced a vision of using ear-worn devices as the *preferred*, mass-market wearable platform, for both (a) individualized, fine-grained monitoring of gym exercise activities, and (b) subsequent real-time, context-aware feedback on exercise dynamics. As exercise-driven motion signals are often too weak to be distinctly captured by a earable device, we propose a novel, hybrid architecture for multi-user gym environments, where joint statistical analysis of equipment-mounted IoT and earable sensor data are used to match individuals to specific gym equipment. We showed that, in spite of the significant signal dampening on the earable, it is possible to extract salient frequency components of exercise-related motion, and that there exists strong correlation (> 0.71) between the relevant earable and equipment features. Early experimental results suggest that such correlation can be used to accurately identify (user, equipment) pairings among current users (83% of such pairings were correctly identified). Furthermore, the combined inertial signals from ear-worn and equipment-mounted sensors can be used to classify exercises (from among 8 distinct choices) with 92% accuracy, a notable improvement over the accuracy achieved from either device alone.

ACKNOWLEDGMENTS

This material is supported partially by the National Research Foundation, Prime Minister's Office, Singapore under its International Research Centres in Singapore Funding Initiative, and partially by the U.S. Army International Technology Center Pacific (ITCPAC), under Contract No. FA5209-17-C-0006.

REFERENCES

[1] Oliver Amft, Mathias Stäger, Paul Lukowicz, and Gerhard Tröster. 2005. Analysis of chewing sounds for dietary monitoring. In *(UbiComp'05)*.

[2] Louis Atallah, Anatole Wiik, Gareth G Jones, Benny Lo, Justin P Cobb, Andrew Amis, and Guang-Zhong Yang. 2012. Validation of an ear-worn sensor for gait monitoring using a force-plate instrumented treadmill. *Gait & posture* 35, 4 (2012), 674–676.

[3] Shengjie Bi, Tao Wang, Nicole Tobias, Josephine Nordrum, Shang Wang, George Halvorsen, Sougata Sen, Ronald Peterson, Kofi Odame, Kelly Caine, et al. 2018. Auracle: Detecting Eating Episodes with an Ear-mounted Sensor. *Proc. of the ACM IMWUT* 2, 3 (2018), 92.

[4] Keng-Hao Chang, Mike Y Chen, and John Canny. 2007. Tracking free-weight exercises. In *(UbiComp'07)*.

[5] Han Ding, Longfei Shangguan, Zheng Yang, Jinsong Han, Zimu Zhou, Panlong Yang, Wei Xi, and Jizhong Zhao. 2015. Femo: A platform for free-weight exercise monitoring with rfids. In *(SenSys'15)*.

[6] Bruno Gil, Salzitsa Anastasova, and Guang Z Yang. 2019. A Smart Wireless Ear-Worn Device for Cardiovascular and Sweat Parameter Monitoring During Physical Exercise: Design and Performance Results. *Sensors* 19, 7 (2019), 1616.

[7] Xiaonan Guo, Jian Liu, Cong Shi, Hongbo Liu, Yingying Chen, and Mooi Choo Chuah. 2018. Device-free Personalized Fitness Assistant Using WiFi. *Proc. of ACM IMWUT* 2, 4 (2018), 165.

[8] Timothy C Havens, Gregory L Alexander, Carmen Abbott, James M Keller, Marjorie Skubic, and Marilyn Rantz. 2009. Contour tracking of human exercises. In *CIVI'09*. IEEE.

[9] Emily Knight, Melanie I Stuckey, Harry Prapavessis, and Robert J Petrella. 2015. Public health guidelines for physical activity: is there an app for that? A review of android and apple app stores. *JMIR mHealth and uHealth* 3, 2 (2015), e43.

[10] Julia S Mollee, Anouk Middelweerd, Robin L Kurvers, and Michel CA Klein. 2017. What technological features are used in smartphone apps that promote physical activity? A review and content analysis. *Personal and Ubiquitous Computing* 21, 4 (2017), 633–643.

[11] Dan Morris, T Scott Saponas, Andrew Guillory, and Ilya Kelner. 2014. RecoFit: using a wearable sensor to find, recognize, and count repetitive exercises. In *(CHI'14)*.

[12] Shahriar Nirjon, Robert F Dickerson, Qiang Li, Philip Asare, John A Stankovic, Dezhi Hong, Ben Zhang, Xiaofan Jiang, Guobin Shen, and Feng Zhao. 2012. Musicalheart: A hearty way of listening to music. In *SenSys'12*.

[13] Md Fazlay Rabbi, Taiwoo Park, Biyi Fang, Mi Zhang, and Youngki Lee. 2018. When Virtual Reality Meets IoT in the Gym: Enabling Immersive and Interactive Machine Exercise. *Proc. of the ACM on Interact. Mob. Wearable Ubiquitous Technol.* 2, 2 (2018), 78.

[14] Statista 2017. [n. d.]. Global unit sales of headphones and headsets from 2013 to 2017 (in millions). ([n. d.]). https://www.statista.com/statistics/327000/worldwide-sales-headphones-headsets/; Accessed: August'19.

[15] Gábor J Székely, Maria L Rizzo, Nail K Bakirov, et al. 2007. Measuring and testing dependence by correlation of distances. *The annals of statistics* (2007).

[16] Eduardo Velloso, Andreas Bulling, Hans Gellersen, Wallace Ugulino, and Hugo Fuks. 2013. Qualitative activity recognition of weight lifting exercises. In *(AH'13)*.

STEAR: Robust Step Counting from Earables

Jay Prakash
University of Illinois at Urbana Champaign (UIUC)
Singapore University of Technology and Design (SUTD)

Zhijian Yang
University of Illinois at Urbana Champaign (UIUC)

Yu-Lin Wei
University of Illinois at Urbana Champaign (UIUC)

Romit Roy Choudhury
University of Illinois at Urbana Champaign (UIUC)

ABSTRACT

This paper shows that inertial measurement units (IMUs) inside earphones offer a clear advantage in counting the number of steps a user has walked. While step-count has been extensively studied in the mobile computing community, there is wide consensus that false positives are common. The main reason for false positives is due to limb and device motions producing the same periodic bounce as the human walk. However, when IMUs are at the ear, we find that many of the lower-body motions are naturally "filtered out", i.e., these noisy motions do not propagate all the way up to the ear. Hence, the earphone IMU detects a bounce produced only from walking. While head movements can still pollute this bouncing signal, we develop methods to alleviate the problem. Results show 95% step-count accuracy even in the most difficult test case – very slow walk – where smartphone and fitbit-like systems falter. Importantly, our system *STEAR* is robust to changes in walking patterns and scales well across different users. Additionally, we demonstrate how *STEAR* also bring opportunities for effective jump analysis, often important for exercises and injury-related rehabilitation.

1 INTRODUCTION

Step-counting has been an important primitive for the mobile/wearable industry, including smartphones, smartwatches, fitbits, arm-bands, etc. Applications have used step-count to derive statistics like calorie burnt, exercise tracking, activity loggers, and even gait analysis for post-injury rehabilitation. Yet, there is wide agreement that step-count is still not accurate; random actions of the leg and hand lead to over/under estimates. For instance, shaking one's leg while seated can increment the step counter, as could playing drums, ping-pong, or video games.

The problem of accurate step-counting with IMUs is non-trivial. Briefly, the human body bounces as a user walks and this bounce manifests into a periodic sinusoidal signal in the IMU's accelerometer. Step count is essentially the frequency (or the number of peaks) of this sinusoidal signal. In reality, however, various other motions of the human body (and the device) get added to the IMU measurements, polluting the sinusoid, or injecting spurious periodicity even when the user is not walking. Hence, the technical challenge lies in continuously identifying and filtering out these pollutions, while robustly adapting to the user's variations in walking patterns. For instance, a user may walk differently during a stroll on a beach, during a walk to the office, while walking down the stairs, or when tipsy from an evening party. Quickly recognizing these unseen patterns, and yet, filtering out false positives, is a difficult problem. Finally, the IMU itself is cheap, hence noisy, adding an extra layer of complexity to signal processing.

With IMUs becoming popular in modern earphones, a natural question is: *does motion tracking in general, and step counting in particular, benefit from IMUs placed at the ear?* This paper finds that while tradeoffs exist, the net outcome is favorable. In particular, the bounce of the human body during a walk gets reflected at the earphone's IMU, but the random motions (of the leg and hand) get filtered out to a large extent. Said differently, the sinusoid from human walking emerges as a cleaner signal when measured from the ear. Of course, the head motions can still pollute the sinusoid, however, the head's movements are generally more constrained, and mostly rotations (as opposed to linear motions). All in all, the net IMU signal lends itself well to signal processing, resulting in robust step-count estimation.

Of course, translating the core opportunity into a robust system entails 2 key challenges. (1) The orientation of the earphone can vary as the user moves her head. If this orientation is not tracked continuously, the IMU data will not project correctly to the global reference frame (explained later), ultimately derailing the step counting system. (2) The bouncing motion varies over different sessions, and even within a session, the signal shape can change. Standard peak detector techniques falter because spurious spikes or fluctuations causes peak counts to get incremented. Filtering the signal around walking frequencies is difficult since the walking frequency is not known *a priori*. Any attempts to predict this frequency makes the system less robust to variations.

In coping with these challenges, our system *STEAR* adopts simple but scalable methods. For orientation, we continuously integrate the gyroscope to compute the 3D projection matrix from the local to the global reference frame. Since human head motion is only rotations, such a gyroscope based scheme is sufficient. Once the 3D orientation is known, the accelerometer data is projected on to the global frame. Then, instead of peak detection, we apply a dynamic time warping (DTW) based scheme to cope with signal shape variations, and potential spurious fluctuations. Of course, some degree of pre and post processing is performed on the signal to ultimately extract the step count of the user.

We implement *STEAR* on Nokia's ESense earbuds [4], embedded with an IMU. The IMU samples at the highest supported sampling rate of 100Hz. We ask 7 users to walk in normal modes. We also ask users to perform activities such as jumps and skips. Our evaluation reveals the following:

• *STEAR* measures very slow steps with $\geq$ 95% accuracy, typically a hard problem on phones and watches.

• Earphones are as good, if not better than smartphones in calculating steps in other modes of walking.

• Earphones measure jumping characteristics better using unique properties like a trail of zero-acceleration while a person is in free fall. Smartphones are subjected to noise due to relative jitters and friction with pants and bags, ultimately affecting the jump analysis.

In sum, the contribution of this paper may be summarized as follows: *we show a natural opportunity that human walking motion is better reflected in earphones, and we design a system to harness this opportunity, ultimately resulting in a robust step-counter and jump detection method for earphones.*

The rest of the paper expands on this core contribution, starting with some ground measurements, followed by system design, evaluation, and conclusion.

2 GROUND MEASUREMENTS

This paper builds on the premise that IMUs on earphones are less impacted by noise and arbitrary limb motions compared to IMUs in smartphones, wrist-bands, and smartwatches. Figure 1 visually illustrates the motion trajectory of the head during a walk – this trajectory mimics a sinusoid. To verify whether a real IMU measures a sinusoid as well, we record IMU acceleration from a earbud and a phone during a normal walk. Their time and frequency representations are shown in Figure 2(a)-2(d). Evidently, earbuds exhibit much cleaner measurements in comparison to a relatively spread-out and noisy spectrum in phone's IMUs. The frequency spectrum also shows a clearer peak for ear IMUs compared to a mix of comparably strong frequencies for phones.

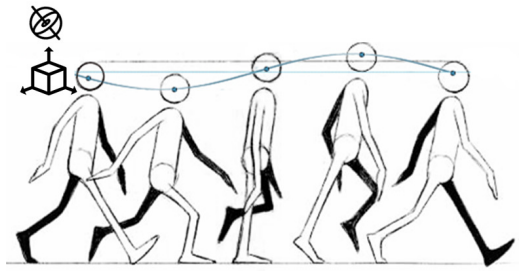


Figure 1: Sinusoidal motion of a head during walk

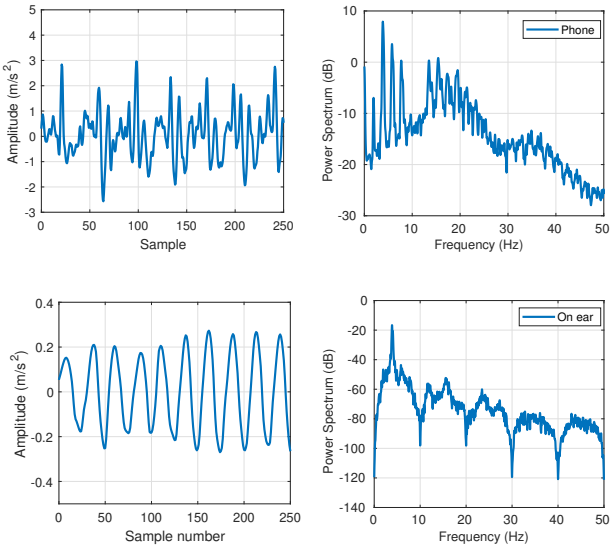


Figure 2: Acceleration (a) as recorded from phone (b) spectrum at phone (c) as recorded from earable (d) spectrum at earable.

Next, we study the IMU's signal-to-noise ratio (SNR) across different body locations and activities. SNR here is defined in terms of the ratio of signal power during a period of activity to the power of the noise during a time segment before the activity. Figure 4 shows the results. The participant walked slowly, normally, and ran for the third session; he wore a earbud on the ear, and carried a smartphone in hand and another in the pocket. Evidently, earbuds exhibit an advantage over the phone in the studied scenarios. These form the basis for a robust physio-analytics framework, developed in the rest of the paper.

Walking Pattern Variations

Figure 5(a) and (b) plot the time domain IMU data from multiple walking sessions – the former is from smartphones while the latter is from earbuds. The multiple lines in each graph are from different users. Clearly, smartphones show more variations, suggesting extraneous pollutions from different parts of the lower body. In contrast, earbuds are consistent,

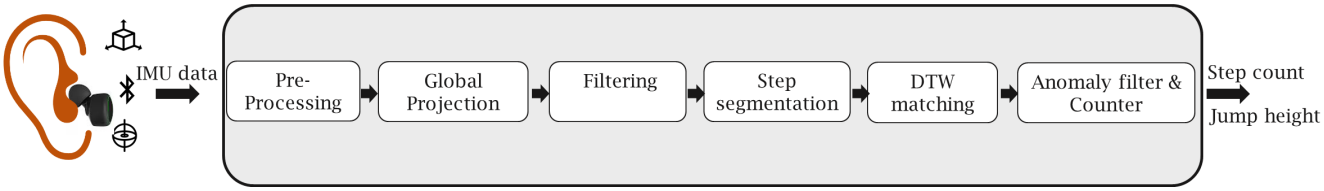


Figure 3: System Architecture for Step Counting

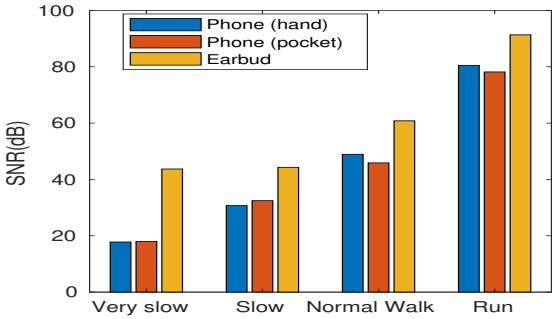


Figure 4: SNR at Esense and phone for different activities

again due to the effects of natural filtering. This further endorses the opportunity of robust step counting with earable IMUs.

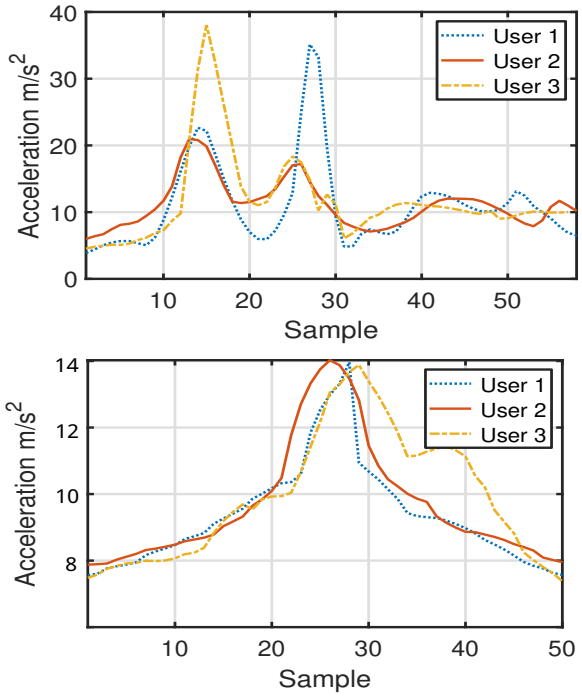


Figure 5: Comparing templates of steps for different users (a) IMU in phone placed in pocket, (b) IMU in earbud.

3 *STEAR*: SYSTEM DESIGN

STEAR has two modules: (1) a step counter that remains always ON, and (2) an exercise analytics module that is activated on-demand. We first describe the pipeline for counting steps, followed by methods for basic exercise, specifically jump analysis. Both these are simple and lightweight, lending themselves well to on-device, real-time processing, even on a small earable platform.

Step Count

Figure 3 shows the processing pipeline for the proposed step counter, composed of the following modules:

■ **Pre-Processings:** Signal pre-processing mainly includes cancelling the bias in the gyroscope. Bias in gyroscope readings gets magnified after integration and affects the projection matrix to be discussed next. *STEAR* identifies when the device is static[1], and during this static window, computes the average of the gyroscope readings. If the readings are only due to (zero-mean) noise, the average should tend to zero. However, if the gyroscope exhibits a bias, it should be revealed here as a DC value. We subtract this average value over subsequent measurements, thereby compensating the bias. Clearly, larger the static window, better is the estimate, and we use many minutes for averaging.

■ **Global Projection:** IMU sensors report readings in its local reference frame. However, to understand motion trajectories like the one in Figure 1, the data needs to be projected to the global framework. This is because as the device moves, the $<X, Y, Z>$ coordinates are constantly changing/rotating, and the IMU measures the acceleration with respect to its *instantaneous* $<X, Y, Z>$ axes. To explain with an analogy, an IMU is like an airplane passenger who is able to tell that the plane is accelerating "forward" or taking a "left" turn (i.e., in her instantaneous $<X, Y, Z>$ axes), but it is hard for the passenger to track the plane's trajectory on the global world map. To track global movement, the IMU's orientation needs to be constantly rotated to keep the plane horizontal and moving North, and the acceleration needs to be measured in

[1]This is not difficult and is performed by checking if the magnitude of the acceleration is same as gravity.

this fixed global orientation. Said differently, the acceleration needs to be projected to this global 3D orientation.

STEAR performs this projection by integrating gyroscope data to estimate global orientation (a standard process in literature). Since gyroscopes drift, we reset it at static instants. As before, static instants are detected when the earphone's acceleration equals gravity and gyroscope measures zero readings (or just noise).

Let's denote the local accelerometer reading at static points as vector $a_{local} = [a_1, a_2, a_3]$. We find a rotation matrix R, which rotates a_{local} to $a_{global} = [0, 0, g]$. Note that R is not unique, because we did not specify the horizontal rotation. However, this will not cause a problem because we only care about vertical movements. After that, for each gyroscope reading at time t_i, we constantly apply delta rotation matrix ΔR_i, representing the delta rotation from time t_i to t_{i+1}, to the original rotation matrix R to give the projection for each time stamp afterwards.

■ **Filtering:** Although earable IMUs offer better SNR, filtering can still be useful. Given that pollutions from some head motions are possible, we can still eliminate them since they are at low frequencies. We apply both low and high pass filter (using moving averages).

■ **Step Segmentation and DTW matching:** Traditional smartphone-based step counting uses peak detection on the accelerometer data. Given the diversity of gaits and smartphone placements, it is hard to separate peaks from walking steps, and those from (unrelated) limb or device movements. This causes errors in today's counters. With head motion, on the other hand, the peaks are cleaner and thereby lends itself to matching against a walking template. *Of course, there is no global template since humans walks with different step lengths, frequencies, and sways. Nonetheless, these variations can be modeled as a compression or expansion of a simple walking template.* This protect the step-counter from detecting spurious peaks and noisy false positives, while being robust to variations in walking patterns. To this end, the measured acceleration is first normalized based on the amplitude, and then matched with the template using a dynamic time warping (DTW) algorithm. DTW accommodates the variations of walks while detecting the peaks quite accurately. Importantly, no training is necessary.

■ **Anomaly (Step or Not) Filter and Counter:** We set a threshold on the DTW score to determine whether a potential step is an actual step or not. The threshold is set as two-thirds the standard deviation of all the DTW scores from a recent period of walking. This is of course a heuristic, however, results show that it adapts to changes in walking patterns. If one intends to reduce false positives (but can tolerate false negatives), the threshold can be increased.

Exercise Analytics: Jump Measurement

We focus on only one instance of exercise analysis: *jumping*. How high a user can jump is an important metric for various health checks [6]. Earphone IMUs offer an opportunity since it is fixed at the ear and does not jiggle/jitter like a phone in a pocket. Even smartwatches show some vibration since it is not tightly worn on the wrist, and the user would move her hands during the jump.

We propose to measure jump-height by observing that the accelerometer reads ZERO when the user is off the ground (see Figure 6). This is because the accelerometer actually measures the reactive force on the body, which is 0 after the feet leave the ground. This yields the exact start and end time of the jump, and applying basic physics and kinematics during this time window, we expect to calculate the height. This avoids the need to do double integrate the accelerometer (which results in a poor estimate due to noise integration).

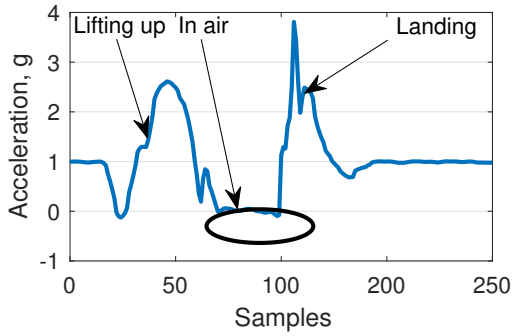


Figure 6: Jumping analysis: 3 stages. Accelerometer reading is 0 when the user is in the air.

■ **Jump Height Calculation:** We detect jumping by setting a threshold ϵ, slightly above zero. When the accelerometer reading is $< \epsilon$, we believe the user is in air. Denoting t as the total time in air, and the rise/fall movement is symmetric in time (i.e., $t/2$), we calculate the jump height simply as $h = 1/8gt^2$, where g is known acceleration due to gravity.

4 IMPLEMENTATION AND EVALUATION

STEAR is built using onboard IMU of an Esense [4] earbud from Nokia. The IMU data stream, sampled at 100 Hz, is recorded through an Android application, nRF Connect by Nordic Semiconductor [7], and piped to MATLAB. The smartphone baseline result is obtained from OnePlus 3T smartphones running 3 most popular step counting apps, downloaded from the Android Play Store.

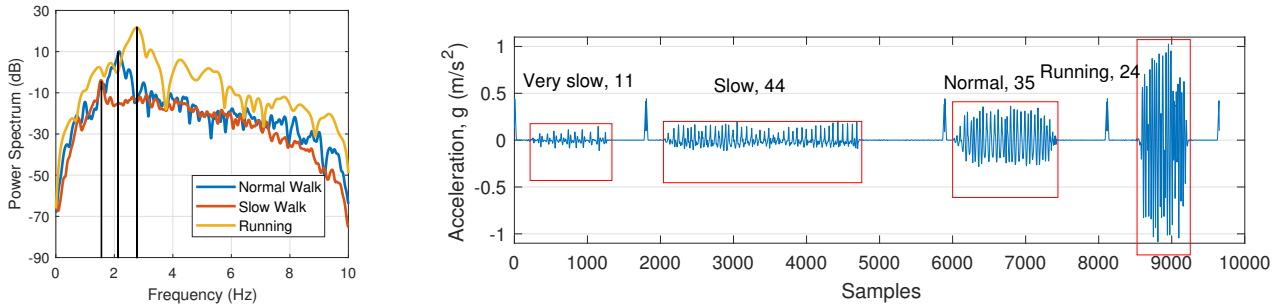


Figure 7: (a) Acceleration spectrum across walking modes. (b) Time domain, filtered, signals for walking modes for same user.

Step Count

Four sets of experiments were performed with the participants wearing Esense on the ear and carrying smartphones in their pockets and hands. Participants were requested to walk in 4 modes: (a) very slow, (b) slow, (c) normal and (d) run. Figure 7(a) shows that the earbud captures the shift in the peak frequency with increasing walking speed. Figure 7(b) also visualizes the time-domain signals, showing the variations possible for the same person.

■ **Very slow walk:** It is crucial to be able to track very slow steps for assisting medical recovery in patients, older adults, and for slow walks inside houses or in beaches. Many prior work [2]-[5] report that traditional equipment and applications are inaccurate for adults walking at a speed of <0.9 m/s. Figure 8 verifies this result – smartphone miss most of the instances of slow walks since the peaks are buried under noise and random limb motions (since the peak amplitudes are weak for slow walks). With *STEAR*, due to of high SNR and nearly sinusoidal observations at earbuds, we are able to achieve higher accuracy in step-count. Figure 8 shows the comparison. We tracked 3 users moving very slowly and taking small steps, emulating walking old adult or patients. *STEAR* achieves accuracy >97% – 49 out of 50 steps – while smartphones under-counts as 5/50.

■ **Slow walk:** Slow walk corresponds to small and low-speed steps which we usually take while moving inside the home or while talking to a friend or while moving in groups. We tracked 3 users moving slowly and taking small steps, emulating such slow walks. We are able to achieve an accuracy of >98% on average. Tracking such steps is important for applications like indoor localization [9], where GPS like capabilities are unavailable, and pedestrian dead-reckoning is a candidate solution.

■ **Normal walk:** A typical walk produces promising SNR which reflects in near-perfect, >99%, accuracy. Of course, this is the reason phones and watches perform quite well since the peaks from each step rise above the noise floor. Several apps are tuned to identify this mode of walking.

■ **Running:** *STEAR* is able to count running steps with high accuracy as well. Figure 7(b) shows that there exists an opportunity of exploitation here, in the form of counting number of $-g + \epsilon$ acceleration points. In other words, when the runner's legs both are off the ground, the IMU accelerations shows an instantaneous ZERO measurement, which precisely counts the number of steps. Thus, large random hardware noise is the only reason to mis-count steps during running.

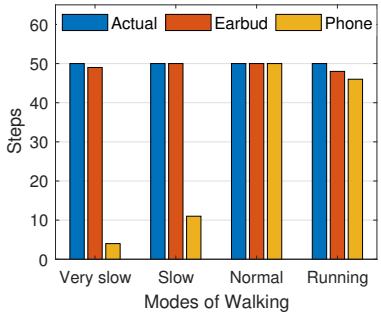


Figure 8: Steps counts under different walking modes

We observe that when a user has a phone in his hand and walks with a typical gait, both earbuds and phones perform equally well, as shown in Figure 9, and achieve near-perfect accuracy. But walking with the phone in pockets – both left-pocket (LP) and right-pocket (RP) – or playing with it, leads to over-estimation of step-counts.

■ **Jumping:** Medical practitioners suggest numerous styles of jumping for recoveries [6]. As shown previously in Figure 6, our evaluation suggests that we can exploit models of jump to find out instants of (a) rising up for the jump (smaller peak), (b) landing back on the ground (larger peak) and (c) time spent in the air (based on zero acceleration). Naturally, it is possible to count number of jumps with near-perfect accuracy. Also, the height of a jump, given by $gt^2/8$, can be calculated from the length of the horizontal line between

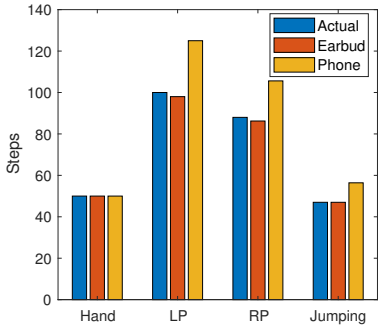


Figure 9: Steps counts under different scenarios

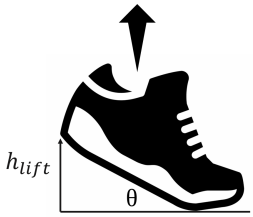


Figure 10: Jump height compensation as per shoe lift

Table 1: Jump height estimation

Trials	1	2	3	4	5
Actual height (cm)	21	20	17	19	20
Calculated height (cm)	20	21	15	17	18

rising and landing. This unique opportunity, a long run of 0 acceleration, can be exploited to identify jump from a series of activities as well. But we needed to compensate for, h_{lift}, the height of the heel when toe lifts off from the ground because acceleration goes to 0 only when the whole body is completely in air (see Figure 10). *STEAR* measures jump height with an error of $+/-2$cm while the count of jumps is accurate to the ground truth, as presented in Table 1.

5 RELATED WORK

Wearables based step counter: Step counters are nothing new and have been implemented on a lot of mobile/wearable devices. The technique is not difficult, but it can suffer from errors. According to a recent measurement study [1], there is an 18.48% error in step counting over a 24-hour free-living period. Our work with an accuracy of $> 95\%$ provides a new opportunity to do better step counting in daily life.

Jump analytics using inertial sensors: There are an abundance of work that tracks various kinds of human motion, [6], MUSE [8] and SensorTape [3], to name a few. The closest to our work is [6], where they also calculate jump height using IMUs. However, the use of double integration to get vertical displacement, is subjected to noise. Our method, which explores the opportunity where acceleration equals to zero, is much more robust.

6 CONCLUSION

This paper shows the promise of robust step counting through ear mounted IMUs in modern earphones. The core opportunity emerges from the observation that the human body serves as a natural "filter", eliminating the noisy movements and only allowing certain walk-related vibrations to propagate up to the ear. We conjecture that this is due to the anatomical structure of the body – the joints in the skeletons and the muscles and tissues – which absorbs higher frequency movements, however, understanding the reasons for such filtering is a topic of study in another field (e.g., kinesiology). We benefit from this natural opportunity by demonstrating that physio-analytics can be improved with earables, not only in robustly counting steps but also in measuring the height of human jumps. Our ongoing investigation is focussed on further improvements to sensing these actions, and exploring other unique motion-related opportunities from earable IMUs.

REFERENCES

[1] Hyun-Sung An, Gregory C Jones, Seoung-Ki Kang, Gregory J Welk, and Jung-Min Lee. 2017. How valid are wearable physical activity trackers for measuring steps? *European journal of sport science* 17, 3 (2017), 360–368.

[2] David R Bassett and Dinesh John. 2010. Use of pedometers and accelerometers in clinical populations: validity and reliability issues. *Physical therapy reviews* 15, 3 (2010), 135–142.

[3] Artem Dementyev, Hsin-Liu Cindy Kao, and Joseph A Paradiso. 2015. Sensortape: Modular and programmable 3d-aware dense sensor network on a tape. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 649–658.

[4] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (2018), 83–89.

[5] Jessica B Martin, Katarina M Krc, Emily A Mitchell, Janice J Eng, and Jeremy W Noble. 2012. Pedometer accuracy in slow-walking older adults. *International journal of therapy and rehabilitation* 19, 7 (2012), 387–393.

[6] Bojan Milosevic and Elisabetta Farella. 2015. Wearable Inertial Sensor for Jump Performance Analysis. In *Proceedings of the 2015 Workshop on Wearable Systems and Applications (WearSys '15)*. ACM, New York, NY, USA, 15–20. DOI:https://doi.org/10.1145/2753509.2753512

[7] Nordic Semiconductor. 2019. nRF Connect for Mobile. (Aug 2019). Retrieved Aug 12, 2019 from https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp&hl=en$_{U}$S

[8] Sheng Shen, Mahanth Gowda, and Romit Roy Choudhury. 2018. Closing the gaps in inertial motion tracking. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. ACM, 429–444.

[9] He Wang, Souvik Sen, Alexander Mariakakis, Romit Roy Choudhury, Ahmed Elgohary, Moustafa Farid, and Moustafa Youssef. 2012. Unsupervised indoor localization. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 499–500.

Towards In-Ear Inertial Jaw Clenching Detection

Siddharth Rupavatharam
WINLAB, Rutgers University
North Brunswick, New Jersey
sidr@winlab.rutgers.edu

Marco Gruteser
WINLAB, Rutgers University
North Brunswick, New Jersey
gruteser@winlab.rutgers.edu

ABSTRACT

Bruxism is a jaw-muscle condition characterized by repetitive clenching or grinding of teeth. Existing methods of detecting jaw clenching towards diagnosing bruxism are either invasive or not very reliable. As a first step towards building a reliable, non-invasive and light weight bruxism detector, we propose an eSense based in-ear inertial jaw clenching detection technique that detects peaks/dips in gyroscope vector magnitude. We also present results from preliminary experiments that show an equal error rate of 1% when the person is stationary and 4% when moving.

ACM Reference Format:
Siddharth Rupavatharam and Marco Gruteser. 2019. Towards In-Ear Inertial Jaw Clenching Detection. In *1st International Workshop on Earable Computing (EarComp'19), September 9, 2019, London, United Kingdom.* ACM, London, UK, 2 pages. https://doi.org/10.1145/3345615.3361134

1 INTRODUCTION

The term *bruxism* is derived from the greek word *brychein* meaning to grind or gnash teeth. More formally, according to F. Lobbezoo et. al. [2], *"bruxism is defined as a repetitive jaw-muscle activity characterized by clenching or grinding of the teeth and/or by bracing or thrusting of the mandible (jaw bone). Bruxism has two distinct circadian manifestations: it can occur during sleep (indicated as sleep bruxism) or during wakefulness (indicated as awake bruxism)"*. Bruxism also does not occur when a person is chewing or swallowing and hence is clearly distinct from jaw movements made to ingest food.

Bruxism affects the quality of life of a person by manifesting in the form of pain in the jaw, face and head, dental problems such as tooth wear and reducing lifespan or wearing down of dental restorations. Present methods of detecting bruxism are through the use of questionnaires , clinical evaluations , inserting intra-oral splints, measuring jaw muscle electromyograph or by polysomnography [3]. Splints, electromyography and polysomnography use sensors that are bulky and invasive and hence influence the jaw clenching and grinding patterns. Questionnaires and clinical evaluations are less accurate because they depend rather heavily on patients' replies to a subjective survey or the detection itself is not entirely reliable. Hence, there is a need for a non-invasive, lightweight and reliable way to detect grinding and jaw clenching as a first step towards a bruxism detection method.

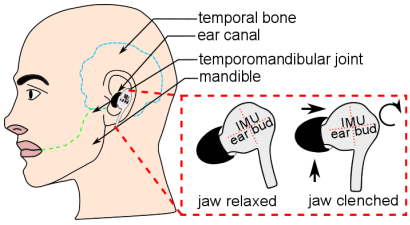


Figure 1: Temporomandibular joint (TMJ) and ear-bud displacement

To address this need, this paper presents an in-ear inertial jaw clenching detection technique and a preliminary evaluation on the eSense platform [1]. In-ear sensing allows jaw clenching data to be collected ambiently and deployment is as easy as using ear-buds to listen to music while performing daily activities.

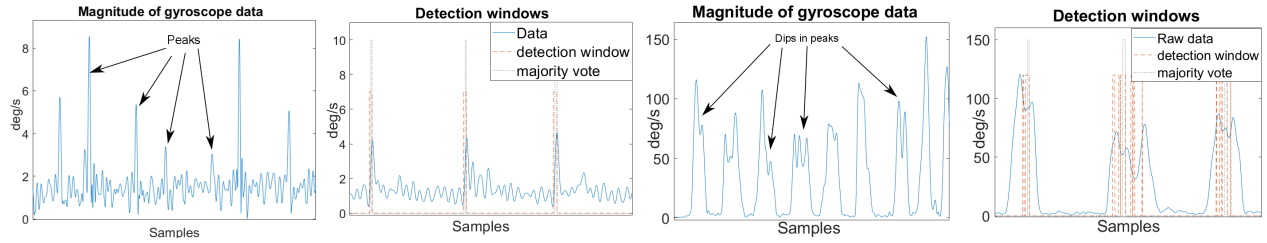
2 DETECTION TECHNIQUE

The observation that enables detection of jaw clenching is that the temporomandible joint (TMJ) is the place where the mandible (jaw bone) meets the temporal bone (on the skull) and the joint is located very close to the ear canal as shown in Fig. 1, hence when the jaw is clenched, the movement of the mandible can be detected through the ear canal. A unique advantage is that both the right and left joints on the jaw bone move together and not independently. Hence, jaw movement can be detected equally in both ears from inside their respective ear canals.

Information regarding the amount of movement of the mandible can be obtained when a person wears ear-buds, such as the eSense, which house an inertial measurement unit (IMU) containing an accelerometer, gyroscope and given that the ear-buds can be inserted sufficiently deep into the

EarComp'19, September 9, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6902-2/19/09…$15.00
https://doi.org/10.1145/3345615.3361134



(a) Gyroscope data - stationary case (b) Detections - stationary case (c) Gyroscope data - mobile case (d) Detections - mobile case

Figure 2: Gyroscope magnitude data and detection of jaw clenching

ear canal. This is because, when the mandible moves up (clenching) and down (relaxing), it creates small depressions in the ear canal which physically displace the ear-bud by small amounts. By observing the amount of displacement and when it occurs, jaw clenching can be detected.

Detection algorithm: Two separate algorithms (peak and dip detection) are used on collected gyroscope magnitude data to detect jaw clenches. Jaw clenches are seen as peaks in gyroscope magnitude data as shown in figure 2(a) in the stationary case. The peak algorithm averages gyroscope magnitude values over a window and compares them to a threshold thr, representing a minimum deg/s. If average window values are greater than thr, a jaw clench is detected as shown in figure 2(b). When the head is in motion, jaw clenches are seen as dips that occur in peaks due to the motion in gyroscope magnitude data as shown in figure 2(c). We explore a separate dip algorithm for this case that averages gyroscope magnitude values in a window and compares them to a threshold window $thrWin$, representing a range (i.e., $upper-lower$ value) in deg/s). If average window values fall within this range, a jaw clench is detected as shown in figure 2(d).

3 EVALUATION

Evaluation is performed for two cases:

(i) Stationary case, when the subject is stationary (i.e., lying down or sitting up but not moving) and only clenches their jaw to mimic sleep bruxism, and **(ii) Mobile case**, when the subjects head is mobile (i.e., working at a desk, looking around, gentle head movements) and clenches their jaw to mimic awake bruxism.

All experiments were performed by a single participant. Each set of experiments was conducted in a window of one minute to reduce IMU sensor drift. The participant was asked to clench his jaws for a duration of 2-3 seconds, every 10 seconds. The experiment was repeated five times, to generate total data of 50 jaw clenches per case. The accelerometer in the ear-buds was calibrated to 0g, 0g and -1g on the X, Y and Z axes respectively and the three gyroscope axes to 0 deg/s by placing on a stationary, flat surface. All collected data was passed through a 2 Hz lowpass filter and then through a peak/dip detection algorithm depending on the case.

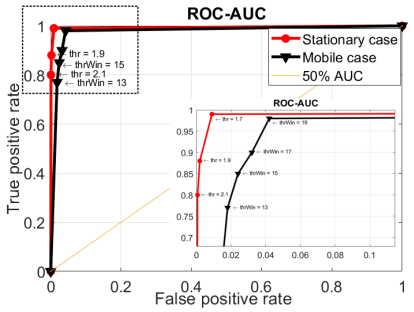


Figure 3: Accuracy in detecting jaw clenching

The detection accuracy is shown in figure 3. The area under curve (AUC) is 99.40% with an equal error rate (EER) of 1% for the stationary case and AUC of 97.66% and EER of 4% for the mobile case.

4 CONCLUSION

In this paper we introduce a novel ear-buds IMU based technique to detect jaw clenching (that might be due to sleep or awake bruxism). In preliminary experiments, the equal error rate of detection is 1% and 4% for a stationary and moving subject, respectively. While these results still need to be validated in a longer study with multiple participants, we believe that these results represent a first step towards a more reliable, non-invasive and light weight bruxism detector through in-ear inertial sensing.

5 ACKNOWLEDGEMENT

We would like to thank Nokia Bell Labs, Cambridge, UK for providing us with the eSense earbuds.

REFERENCES

[1] Kawsar, F., Min, C., Mathur, A., and Montanari, A. Earables for personal-scale behavior analytics. *IEEE Pervasive Computing 17*, 3 (2018), 83–89.

[2] Lobbezoo, F., Ahlberg, J., Glaros, A., Kato, T., Koyano, K., Lavigne, G., De Leeuw, R., Manfredini, D., Svensson, P., and Winocur, E. Bruxism defined and graded: an international consensus. *Journal of oral rehabilitation 40*, 1 (2013), 2–4.

[3] Manfredini, D., and Lobbezoo, F. Relationship between bruxism and temporomandibular disorders: a systematic review of literature from 1998 to 2008. *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology, and Endodontology 109*, 6 (2010), e26–e50.

eSense Veers: A Case Study of Acoustical Manipulation in Walking without Sight both on Subtle and Overt Conditions

Kohei Matsumura*
Kazushi Okada*
matsumur@acm.org
kokada@mxdlab.net
College of Information Science and Engineering, Ritsumeikan University
Kusatsu, Shiga

ABSTRACT

This paper seeks to explore the possibilities in earable computing with a case study of acoustical manipulation in a walking blindfold scenario. In human locomotion, veering often occurs while walking, especially within the absence of visual cues. We investigated the effect of acoustical manipulation with eSense on both Subtle and Overt conditions by conducting a series of experiments. The results showed that our acoustical manipulation reduced deviations in walking straight in the case of both Subtle and Overt conditions. We highlight one future direction for earable computing.

CCS CONCEPTS

• **Human-centered computing** → *Auditory feedback*; *Ubiquitous and mobile devices*; *Laboratory experiments.*

KEYWORDS

acoustical manipulation, redirected walking, earable

ACM Reference Format:
Kohei Matsumura and Kazushi Okada. 2019. eSense Veers: A Case Study of Acoustical Manipulation in Walking without Sight both on Subtle and Overt Conditions. In *1st International Workshop on Earable Computing (EarComp'19), September 9, 2019, London, United Kingdom.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3345615.3361135

*Both authors contributed equally to this research.

EarComp'19, September 9, 2019, London, United Kingdom

ACM ISBN 978-1-4503-6902-2/19/09…$15.00
https://doi.org/10.1145/3345615.3361135

1 INTRODUCTION

People often wear earbuds while sitting on a chair, walking on the street, or even driving a car. Thanks to the advancements in technologies, earbuds are becoming smarter. Some can connect with a smartphone wirelessly, have microphones to communicate with a virtual assistant (e.g., Siri, Alexa, and Cortana), have touch detection to control their functionalities, and have an embedded proximity sensor to detect whether a user placed the earbud into their ear or not.

The concept of earable computing is beyond that. The concept assumes earbuds (and other earable devices) are wearable computers that have built-in sensing functionalities as well as processing capabilities. eSense is one of the tangible implementations of an earable device [5]. Here, a 6-axis IMU, a microphone, a battery, and speakers are embedded into earbuds with wireless connectivity. Each earbud is as small as commercial ones. Since eSense has a 6-axis IMU (3-axis acceleration and 3-axis rotation), it enables us to measure the head movements of a user and to utilize them for user interaction.

To seek the possibilities of earable computing, we tackle acoustic manipulation when walking blindfolded. In human locomotion, veering is often associated with walking in the dark, in heavy rain/snow or in a crowd. That is to say, veering occurs when visual cues are absent as in cases of reduced visibility. Kallie et al. addressed whether the source of veering in the absence of visual and auditory feedback was better attributed to errors in perceptual encoding or to undetected motor errors [4].

We assume that we can change the walking direction of a walker by playing a cue such as three-dimensional audio with earable devices. There are few studies on acoustic manipulation while walking blindfolded. Millar studied the effects of sound and posture cues on veering [7]. Feigl et al. found that applying a loud noise could make people veer [1].

Although these mainly lean toward visual stimulation, the techniques that manipulate the walking route in the virtual environment are called redirected walking (RDW) [8].

Suma et al. presented a taxonomy that categorizes redirection techniques according to their geometric flexibility versus the likelihood that they will be noticed by users [11]. The techniques are divided into two types of noticeability to the user: Subtle and Overt.

In this paper, we further investigate the effect of acoustical manipulation with both Subtle and Overt conditions, according to Suma et al.'s taxonomy [11]. In order to investigate the effects of acoustical manipulation, we conducted a series of experiments. We also discuss the future direction of earable computing.

2 RELATED WORK

The concept of manipulation of direction while walking is related to redirected walking (RDW) [8], which allows users of virtual reality applications to explore virtual environments larger than the available physical space. This is achieved by manipulating users' walking trajectories through visual rotation of the virtual surroundings, without users noticing this manipulation [9].

Since virtual reality technologies with head-mounted displays (HMD) have become widespread at the consumer level, a number of studies on RDW have been conducted. Razzaque et al. interactively rotated the virtual scene as the user walked about in the real world with an HMD. They reported that RDW causes people to change their real walking direction without noticing it [8]. Ishii et al. used visual processing on images displayed on a HMD. The system consists of a camera and a HMD and works as a video see-through. The system superposes a visual illusion onto the raw video images to manipulate walking direction. Their experiment revealed that their system could change the walking paths of participants by approximately 200 mm/m on average [3].

The intensity of veering manipulation is bounded by its corresponding detection threshold of humans. Steinicke et al. and Grechkin et al. reported the radii of threshold curvatures as 22 m and 5 m, respectively [2, 10].

Matsumoto et al. used visio-haptic multimodal interaction by employing a curved physical wall on the side of a user and revealed that they could reduce the radius of threshold curvatures as 2.5 m [6]. Their work showed possibilities that multimodal manipulation such as visual and tactile, or visual and audio, can reduce the radius of threshold curvatures in RDW.

Although visual stimulus causes people to change their real walking direction and multimodalities have the potential to increase the effectiveness in RDW, there are fewer studies on acoustical manipulation while walking blindfold. Millar studied the effects of sound and posture cues on veering from the target with young blind children [7]. Feigl et al. found that applying a loud noise can make people veer [1].

We further seek the effect of acoustical manipulation in this paper. We use eSense, wearable smart earbuds that are capable of sensing the orientation of the user's head, and play a sound while people are walking blindfold, with both Subtle and Overt conditions (c.f. [4]) to manipulate the direction of walking.

3 ACOUSTICAL MANIPULATION WITH ESENSE

To investigate the effect of acoustical manipulation in walking blindfold on both Subtle and Overt conditions, we implemented a simple audio feedback system. The system plays two types of sounds with a three-dimensional audio technique.

A user of the system wears the earbuds on both ears and hears three-dimensional sounds via the earbuds. The three-dimensional sound is played via a virtual speaker. That is, if the virtual speaker is placed on the right side of the user, the user will notice that the audio is played from the right side.

Figure. 1 shows the position of our virtual speaker. We placed the virtual speaker at a 1.0 m distance from the user's head. Feigl et al. used a virtual speaker technology as well and reported that a 1.0 m distance is optimal [1]. First, the speaker will be located straight ahead of the user's head. When hir head rotates clockwise with angle θ, the speaker will keep its distance with 1.0 m, but it will be located with $-\theta$ $(counterclockwise)$ from the direction of the face (see Figure. 1). Formally, the location of the speaker can be calculated with $speaker_x = rcos\theta$, $speaker_y = rsin\theta$ where r is radius $(= 1.0m)$ from the origin (i.e., position of the head), θ is rotation of the head (clockwise), and $speaker_x$ and $speaker_y$ are calculated position of the speaker. Figure. 2 indicates a sample route and calculated positions of the virtual speaker.

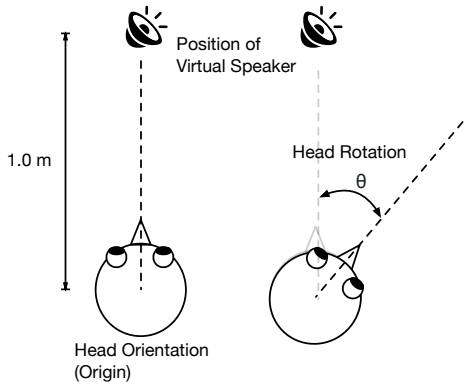


Figure 1: Acoustical manipulation with a virtual speaker. The virtual speaker firstly located at 1.0 m ahead of the origin of the head.

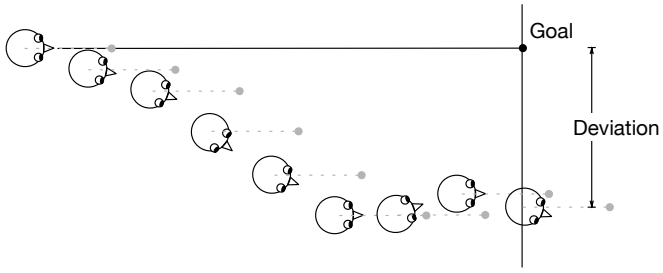


Figure 2: The virtual speaker (indicated as a gray circle with a dotted line) firstly placed at the front of the users head. The position of the speaker will change according to the rotation of the user's head.

Figure 3: A person wearing a blindfold, earbuds, and eSense.

Implementation

To implement the system, we used a blindfold, a pair of wireless earbuds (Powerbeats Pro), eSense [5] and a PC (Macbook Pro) as shown in Figure. 3. Readers may wonder why we use the earbuds despite the fact that eSense has its own speakers. In our setup, the audio functionalities of eSense did not work well. It caused noise or sometimes disconnected, so we used the earbuds for playing sound.

eSense senses rotations of the user's head and sends them to the PC via BLE communication. We 3D-printed an enclosure for eSense to attach eSense to the user's head. The enclosure has two holes for tight attachment to the blindfold's elastic band.

The PC wirelessly connects to both the earbuds and eSense. It receives six-axis inertial data from the eSense, including rotational movements, and plays three-dimensional sound calculated with the abovementioned algorithm according to the rotation of the user's head.

We use the Node.js platform for software on the PC to communicate with eSense and to play two types of acoustic stimuli.

Stimuli: Subtle and Overt

We prepared two types of acoustic stimuli for both Subtle and Overt conditions according to the taxonomy of Suma et al. [11].

For the Subtle condition, we simply played music on our platform. We used *"Walk this way"* by Aerosmith as the stimulus for the Subtle condition. Since the duration of the music was 3:31, the system put it on repeat so that the user would keep hearing it even if ze took time to complete a task in the experiment.

For the Overt condition, we composed music that lasted 1:00 with the digital audio workstation software, Garageband. The music consisted of periodical "Delicate Bell," a pre-installed musical instrument on Garageband, with a note of A4 sounds. We set the time interval of each note to two seconds (i.e., 0.5 Hz).

4 EXPERIMENT

We conducted two experiments to verify if our acoustical manipulation decreased deviations in walking straight. The error is the difference between the position of the goal and the position when a participant walked a predefined distance. For example, if a participant was asked to walk 20 m and veered 5.0 m right from the goal when ze walked 20 m, the error would be 5.0 m.

We experimented with the effect of acoustical manipulation in walking in both subtle (experiment 1) and overt (experiment 2) conditions. The details are described as follows.

Experiment 1: Subtle Condition

We firstly conduct an experiment with a subtle condition. In this condition, we do not explain the purpose of the study nor any technical detail of the system. That is, we expect that none of the participants will notice there is intervention or control over the music played through the earbuds.

Procedure. We ask participants to wear a blindfold with an eSense and a set of earbuds and to walk as straight as possible. The participant walks 50 m unless ze deviates 10 m from the centerline.

We put two sets of belt stanchions just behind the start point to give the participant the direction of the goal. Each set of the stanchion is connected with a belt to ensure that the participant feels the aimed direction. Figure. 4 shows a setup of the experiment.

Before a participant starts walking, we play the music *"Walk this way"* as the stimulus for the subtle condition.

As mentioned before, when a participant deviates 10 m from the centerline before ze reaches the goal line (i.e., 50 m), one of the experimenters gently notifies hir to stop walking.

To compare the difference between the two conditions—the control condition and the experimental condition—we ask each participant to walk three times for the control condition and three times for the experimental condition, six times in total. In the control condition, we simply play the original music on our system. That is, the music will be delivered to both ears of the subject equally. In the experimental condition, our system plays the music according to the algorithm described in the Section. 3. The order of the conditions is randomized. We do not notify the order of the conditions to the participants in the experiment.

We conduct the experiment on the outdoor field. We employ two video cameras and a range finder (Bosch GLM50) to make a record of the experiment. When a participant reached the goal line as Figure. 4-A, we measure the deviations from the center of the goal using the range finder and record them. If the participant deviates more than 10 m from the centerline as Figure. 4-B, we record the length from the start line and calculate the estimated deviation from the goal. To estimate the deviation, we firstly calculate θ with $\theta = \tan^{-1}(d_{10}/10)$ where d_{10} is the distance from the start line. We then calculate the deviation with $d = 50/\tan\theta$ where d is the deviation.

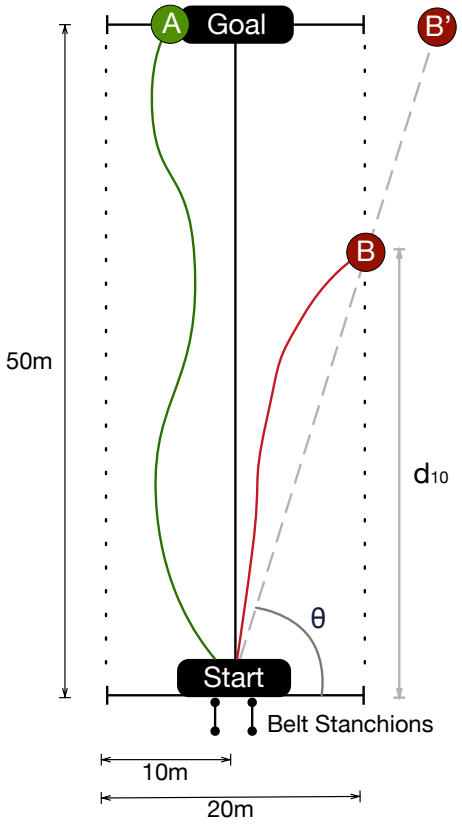


Figure 4: Experimental settings for the experiment 1.

Experiment 2: Overt Condition

We also have an experiment with an overt condition. In this condition, we explain the details of the system to let the participants understand the meaning of the sound. We explain that the bell sounds will play every two seconds, and the position of the virtual speaker moves according to the walking direction.

Procedure. The procedure of experiment 2 is almost the same as experiment 1. The differences are (1) Instructions to participants, (2) Acoustical stimulus, and (3) Experiment field.

As we described previously, we explain the detail of the system to the participant. The participants understand that the position of the virtual speaker will change according to the direction of hir walking direction. We had a short practice session before starting the experiment to get hir used to the system.

For the acoustical stimulus, we used periodical "Delicate Bell" sounds with the 0.5 Hz time interval. It is like a radar sound. We expected that each participant would be aware of the hir walking direction and fix the direction correctly.

Instead of the outdoor field, we experimented in the gymnasium, primarily to protect the participants from heatstroke. Because of the small size of the gymnasiums, the participants walked as straight as possible for 35 m. As with experiment 1, when a participant deviated 10 m from the centerline before ze reached the 35 m line, we notified hir to stop walking.

5 RESULTS

Experiment 1

Seven students (all males) gave written informed consent and participated in the experiment. Each of them walked three times under the control condition and three times under the experimental condition. The order of the conditions was randomized.

Figure. 5 showed an actual image of the experiment. We conducted the experiment on the outdoor field.

The means of deviations on the control condition for each participant are 9.72 m, 6.01 m, 6.81 m, 8.84 m, 10.58 m, 6.15 m and 19.42 m.

The means in the experimental(Subtle) condition are 5.19 m, 1.38 m, 6.06 m, 9.39 m, 6.39 m, 5.52 m, and 17.10m, respectively. Except for participant 4 (he scored 8.84 m on the control condition and 9.39 m on experimental condition), the means of deviation on experimental condition were lower than one on the control condition.

Figure. 6 shows the boxplot for each condition. The dots show the scores (i.e., deviations) for one trial. The means for all trials for both control and experimental conditions are 9.66 m (sd=7.09) and 7.11 m (sd=6.73) respectively. We also tested the statistical difference between two conditions with

Figure 5: An actual image of the participant walkikg in experiment 1.

Figure 7: An actual image of the participant walkikg in experiment 2.

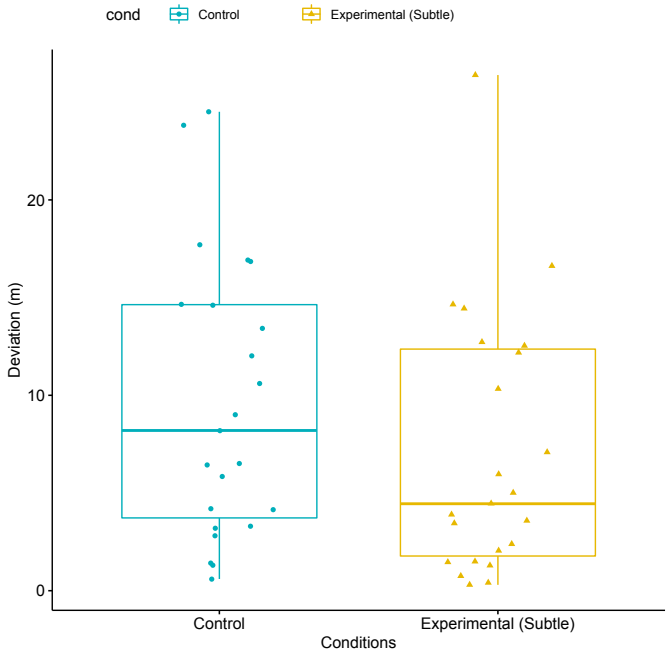


Figure 6: A boxplot for the experiment 1. The box indicates mean, and 1st and 3rd quartiles.

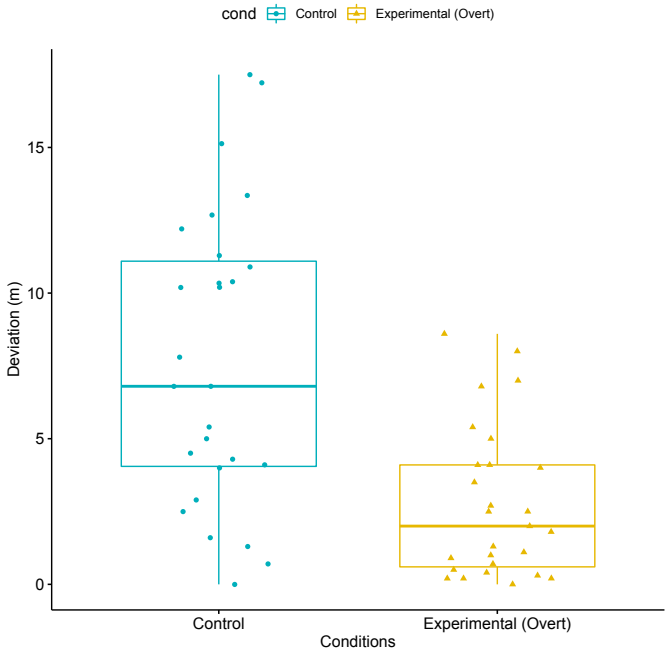


Figure 8: A boxplot for the experiment 2. The box indicates mean, and 1st and 3rd quartiles.

a one-sided paired t-test on 5% significance level. Through the test, we found that the difference between control and experimental conditions greater than 0 ($p = 0.0465$). This means the acoustical manipulation on Subtle condition lowers the deviation on walking blindfold.

We informally interviewed the participants after the experiment. All participants reported that they did not notice that the position of the speaker moved according to the direction of your walking while listening to music.

Experiment 2

Nine students (all males) gave written informed consent participated in the experiment. Note that four of them participated in experiment 1 too.

The means of deviations on the control condition for each participant are 10.91 m, 7.50 m, 4.27 m, 1.80 m, 6.73 m, 12.20 m, 5.03 m, and 10.68 m. Readers may wonder why the deviation on the control condition in experiment 2 is lower than one on the experiment 1. We assume the reason that the walking distance in experiment 2 was shorter than one in experiment 1.

The means of deviations on experimental (Overt) condition are 4.23 m, 0.37 m, 2.03 m, 0.67 m, 2.53 m, 1.40 m, 1.93 m and 6.40 m. You can quickly find that the means of deviation on the experimental condition are lowered than the control condition.

Figure. 8 shows the boxplot for each condition in experiment 2. The dots show the scores (i.e., deviations) for one trial. The means for all trials for both control and experimental conditions are 7.74 m (sd=5.07) and 2.77 m (sd=2.59) respectively. We also tested the statistical difference between two conditions with a one-sided paired t-test on 5% significance level. We found that the difference between control and experimental conditions was greater than 0 ($p < 0.0001$). The 95 percent confidence interval was from 3.211423 to infinity. As with experiment 1, acoustical manipulation on Overt condition also lowers the deviation on walking blindfolded.

6 CONCLUDING REMARKS

In this paper, we investigated the effects of acoustical manipulation while walking blindfolded both on Subtle and Overt conditions. We conducted a series of experiments to verify whether or not our acoustical manipulation decreased deviations in walking straight. Seven and nine students participated in experiment 1 and 2, respectively. In both the Subtle and Overt conditions, statistical tests revealed that there are statistical differences between the control and experimental conditions. It means the acoustical manipulation works on both Subtle and Overt conditions.

The results of the series of the experiment showed the possibilities of earable computing. For example, quite a few people fix their eyes on their smartphones even when they are walking. Thus, they are not aware of their surroundings while gazing at their smartphones, which means they can collide with others. Many people also choose to wear earbuds. If a smartphone is smart enough to sense a person's surroundings and the earbuds have a sense of rotation, the buds can acoustically manipulate the owner and help avoid a collision.

Through the study, we found that we could acoustically manipulate people while they were walking in both the Subtle and Overt conditions. However, we need to compare the effects on both conditions to understand the acoustical manipulation further. We will experiment to compare on both Subtle and Overt conditions with same condition (i.e., at the gymnasium) and will analyze on the difference between the conditions. We also need to make improvements to our system. Currently, the position of the virtual speaker does not reflect the goal position; instead, the position is moved according to the direction of a user's head.

ACKNOWLEDGMENTS

We thank all participants who walked a lot in extremely hot day.

REFERENCES

[1] Tobias Feigl, Eliise Kõre, Christopher Mutschler, and Michael Philippsen. 2017. Acoustical Manipulation for Redirected Walking. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology (VRST '17)*. ACM, New York, NY, USA, Article 45, 2 pages. https://doi.org/10.1145/3139131.3141205

[2] Timofey Grechkin, Jerald Thomas, Mahdi Azmandian, Mark Bolas, and Evan Suma. 2016. Revisiting Detection Thresholds for Redirected Walking: Combining Translation and Curvature Gains. In *Proceedings of the ACM Symposium on Applied Perception (SAP '16)*. ACM, New York, NY, USA, 113–120. https://doi.org/10.1145/2931002.2931018

[3] Akira Ishii, Ippei Suzuki, Shinji Sakamoto, Keita Kanai, Kazuki Takazawa, Hiraku Doi, and Yoichi Ochiai. 2016. Optical Marionette: Graphical Manipulation of Human's Walking Direction. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 705–716. https://doi.org/10.1145/2984511.2984545

[4] Christopher S. Kallie, Paul R. Schrater, and Gordon E. Legge. 2007. Variability in stepping direction explains the veering behavior of blind walkers. *Journal of Experimental Psychology: Human Perception and Performance* 33, 1 (feb 2007), 183–200. https://doi.org/10.1037/0096-1523.33.1.183

[5] Fahim Kawsar, Chulhong Min, and Akhil Mathurand Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 03 (jul 2018), 83–89. https://doi.org/10.1109/MPRV.2018.03367740

[6] Keigo Matsumoto, Yuki Ban, Takuji Narumi, Yohei Yanase, Tomohiro Tanikawa, and Michitaka Hirose. 2016. Unlimited Corridor: Redirected Walking Techniques Using Visuo Haptic Interaction. In *ACM SIGGRAPH 2016 Emerging Technologies (SIGGRAPH '16)*. ACM, New York, NY, USA, Article 20, 2 pages. https://doi.org/10.1145/2929464.2929482

[7] Susanna Millar. 1999. Veering Re-Visited: Noise and Posture Cues in Walking without Sight. *Perception* 28, 6 (jun 1999), 765–780. https://doi.org/10.1068/p2876

[8] Sharif Razzaque, Zachariah Kohn, and Mary C. Whitton. 2001. Redirected Walking. In *Eurographics 2001 - Short Presentations*. Eurographics Association. https://doi.org/10.2312/egs.20011036

[9] Yannick Rothacher, Anh Nguyen, Bigna Lenggenhager, Andreas Kunz, and Peter Brugger. 2018. Visual capture of gait during redirected walking. *Scientific Reports* 8, 1 (dec 2018), 17974. https://doi.org/10.1038/s41598-018-36035-6

[10] Frank Steinicke, Gerd Bruder, Jason Jerald, Harald Frenz, and Markus Lappe. 2010. Estimation of Detection Thresholds for Redirected Walking Techniques. *IEEE Transactions on Visualization and Computer Graphics* 16, 1 (Jan 2010), 17–27. https://doi.org/10.1109/TVCG.2009.62

[11] Evan A. Suma, Gerd Bruder, Frank Steinicke, David M. Krum, and Mark Bolas. 2012. A taxonomy for deploying redirection techniques in immersive virtual environments. In *2012 IEEE Virtual Reailty Workshops (VRW)*. 43–46. https://doi.org/10.1109/VR.2012.6180877

Using an in-ear wearable to annotate activity data across multiple inertial sensors

Alexander Hölzemann*
Univserity of Siegen,
Ubiquitous Computing
alexander.hoelzemann@uni-siegen.de

Henry Odoemelem
Univserity of Siegen,
Ubiquitous Computing
henry.odoemelem@student.uni-siegen.de

Kristof Van Laerhoven
Univserity of Siegen,
Ubiquitous Computing
kvl@eti.uni-siegen.de

ABSTRACT

Wearable activity recognition research needs benchmark data, which rely heavily on synchronizing and annotating the inertial sensor data, in order to validate the activity classifiers. Such validation studies become challenging when recording outside the lab, over longer stretches of time. This paper presents a method that uses an inconspicuous, ear-worn device that allows the wearer to annotate his or her activities as the recording takes place. Since the ear-worn device has integrated inertial sensors, we use cross-correlation over all wearable inertial signals to propagate the annotations over all sensor streams. In a feasibility study with 7 participants performing 6 different physical activities, we show that our algorithm is able to synchronize signals between sensors worn on the body using cross-correlation, typically within a second. A comfort rating scale study has shown that attachment is critical. Button presses can thus define markers in synchronized activity data, resulting in a fast, comfortable, and reliable annotation method.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Computing methodologies** → *Machine learning*.

ACM Reference Format:
Alexander Hölzemann, Henry Odoemelem, and Kristof Van Laerhoven. 2019. Using an in-ear wearable to annotate activity data across multiple inertial sensors. In *EarComp 2019, 1st Interational Workshop on Earable Computing*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/0000000.000000000

*

EarComp 2019, September 10, 2019, London, UK

ACM ISBN 978-1-4503-6317-4/19/07.
https://doi.org/10.1145/0000000.000000000

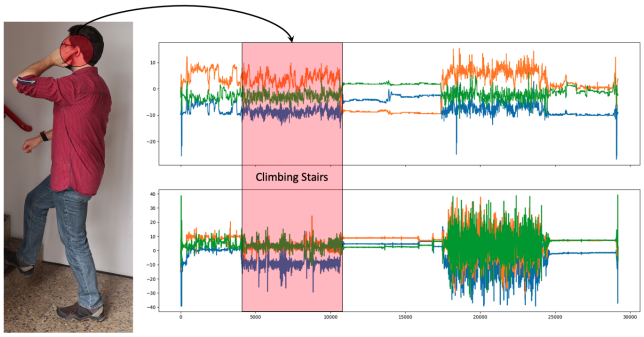


Figure 1: We present an inconspicuous annotation method, in which users can annotate their activity data *in situ* with an in-ear wearable (left), to mark and synchronize inertial data from the ear with all other inertial sensors (right).

1 INTRODUCTION

As wearable sensors have been shrinking and getting less power-hungry, their operation time and places where they can be worn have inadvertently increased accordingly. Nowadays, multiple such sensors can be worn as patches or miniature straps anywhere on the limbs, torso, or even on the head. When doing experiments with such sensor data, however, the annotation of the data has remained a burden, taking a substantial amount of effort. Few methods exist that allow the sensor data to be annotated directly, even fewer methods allow these annotations to be made for any amount of wearable sensor data from the user's body. In this paper, we argue that an in-ear device that is equipped with inertial sensors and a button would be an excellent candidate for user annotation of activity data. It would allow the users to annotate their data without much effort in a socially comfortable way, which also enables 'in the wild' experiments as study volunteers annotate activities in their daily lives. A critical step in our method is the synchronization of sensor data between all wearable sensors: We assume that all sensors contain inertial sensors that show sufficient correlation during everyday activities. The synchronization of different sensor signals plays a decisive role in activity recognition. In most cases, a synchronization gesture is executed at the beginning and end of the measurements to synchronize the

two or more time series. This method has the decisive disadvantage that it is time-consuming and error-prone. With this paper we would like to introduce another approach, that helps synchronizing an arbitrary number of sensor signals. These signals only have the basic preconditions that they must be recorded at the same time and that sufficient sedentary phases, greater than 1 minute, are included. The here presented algorithm works with a very few calculation steps, these are the calculation of the vector length, the standard deviation and a binary filter that is used to decide if the acceleration signal represents a sedentary or non-sedentary activity. The given results show an median time mismatch of 1.10 seconds and can be used to synchronize related, but independently captured, sensor signals with a shared time base. In order for the algorithm to work reliably with the raw data, they must first be prepared and preprocessed. Chapter 4 describes the algorithm in detail. Our presented algorithm is fast and easy to implement. This allows researchers to take up this idea and incorporate it into their projects [2].

2 RELATED WORK

When experiments in human activity recognition rely on multiple inputs from an arbitrary number of sensors, a significant hurdle is to synchronize all sensors' data streams in order to attach them to the same time base. Problems that are typical in such cases have been discussed in detail in previous publications, for example in [1] or [6]. Thus far, several works have been published that deal with the synchronization of two or more independently working sensors. Some of these papers aim at explicitly synchronizing the clocks or time stamps and consider distributed systems or wireless networks, e.g. [14], [16], [15]. A large amount of relevant work is also mentioned in patents, [17], [8], [7], [5], [9] or [19].

[19] describes a procedure that synchronizes the recordings of several media devices with each other [19]. However, the sample applications highlighted in this patent refer to audio signals, such as sound recordings of various musical instruments or vocals. The published system is a client-server application and works with manual set markers in the audio signals. Much of the related research is modality-specific. One approach is presented in [5], where the proposed synchronization technique works with markers that need to bet set in the data. The data then gets synchronized based on these markers. Hesch et al. [7] provide a method that uses a set of interrupt triggered markups. In this system, a processor working in parallel to the CPU is responsible for managing this. [18] developed an algorithm with which he was able to synchronize data coming from a network of seismic sensors. The approach calculates the most probable clock offset for the data. The probably most usable global time was determined from all available sensor clocks by calculating the probability distribution of the clock offset measurements. The most probable offset has been chosen as the offset to tie all clocks to.

Work that is situated in wearable activity recognition research encompasses [20], which presents an approach to achieve robust active learning and avoid the typical annotations errors that asks users to solve a relayed related task and and estimates confidence scores from crowd sourcing.

Kunze et al. [13] published 2006 a method that used features calculated for a sliding window to recognize the sensor position on the human body. The experiment is divided into 5 phases. Firstly the walking activity has been recognized frame by frame. To be sure that only the clean walking segments are used for location recognition, only these frames where more then 70% of the data has been recognized as walking were taken into account. The final total accuracy is 82% and shows that it is possible to create a certain context awareness. This context awareness would be useful in the following investigations for the algorithm presented here. In this way the results could be improved.

3 SYSTEM DESIGN

Hardware

The hardware used for collecting labeled data is the eSense-BLE [11] by the Pervasive Systems group at Nokia Bell Labs, Cambridge. It is built with a custom-designed 15 x 15 x 3 mm PCB and composed of a Qualcomm CSR8670, a dual-mode Bluetooth audio system-on-chip (SoC) with a microphone per earbud; a InvenSense MPU6500 six-axis inertial measurement unit (IMU) including a three-axis accelerometer, a three-axis gyroscope, and a two-state button; a circular LED; associated power regulation; and battery-charging circuitry. It is powered by an ultra-thin 40-mAh LiPo battery, but lacks internal storage or real-time clock. Each earbud weights 20g and is 18 x 20 x 20 mm. The left earbud is the one containing the IMU sensor accessible through the BLE and will be used in the remainder of this paper.

The Platypus prototype is a wrist-worn activity sensing platform [10] that is equipped with a number of sensors, including a full MPU9250 IMU, environmental sensors, and several processing units included in an Edison System-on-Chip module that runs an embedded Linux distribution as operating system. We have used this prototype as it can record the IMU data at a relatively high sampling speed of 300 Hz and present the recordings via a Secure Shell (SSH) over the built-in WiFi transceiver.

Mobile Application

To be able to get labeled data for cross correlation, we developed an Android App for data collection on Android Studio by adapting only the needed aspects of the Android library

provided by the developers. The Android ScanFilter was used to restrict the scan result to our desired eSense device, using the LOW LATENCY scan mode. Notification was enabled by writing to the descriptor for the push button status and accelerometer data from the on-board IMU, which we set to 50Hz sampling rate. The accelerometer works with the default configuration of +/-4g(sensitivity of 8192 LSB/g). Using the Android onCharacteristicChanged, accelerometer data about three-axis is received and checked for correctness using the CheckSum, then stored in the internal storage of our mobile phone as a CSV file in unit of g and multiplied by 10 to increase amplitude. We also saved a time-stamp in microseconds that have elapsed since January 1, 1970 at 00:00:00 GMT. Additional, on every button push, the current data from the accelerometer is labeled with an ASCII character and stored, as well as displayed on the TextView. We had a challenge of receiving the same data on different time stamps, but this was resolved by keeping the processing time in the onCharacteristicChanged method as low as possible, another problem that we encountered, was that each button push notification caused some accelerometer package index to be skipped on subsequent readings, restarting the IMU sampling on each button push solved this problem. Finding and establishing connection with the eSense (BLE and classic Bluetooth) is a challenge and requires several trials.

4 METHODOLOGY

Beside of our study about the reliability of our proposed algorithm we also asked the participants to fill out a short questionnaire regarding the wearing comfort of the eSense earbud by using the Comfort Rating Scale (CRS) as proposed in [12].

Data Set

A data set of activity data of seven participants has been recorded using the eSense and the platypus. The data recorded by the eSense is sampled with 50 Hz, the data recorded with the platypus is sampled with 300 Hz. The participants are between 20 and 40 years old. We were able to recruit five men and two women for the study. The platypus data set consists of a total amount of round about 2.255.764 samples or 2.08 hours of data. For the eSense we recorded 375.967 samples, which also results in 2.08 hours. The data set contains acceleration data from both sensors for mixed activities: (1) read or desk work, (2) walk, (3) climbing stairs, (4) sit, (5) dribble a basketball and (6) pause or rest phase.

eSense Wearing Comfort

In addition to the evaluation of the reliability of our presented algorithm, it was also very important for us to evaluate how comfortable the provided prototype was perceived by the participants of the study. Table 1 shows the categories and their description. The Emotion, Harm, and Anxiety categories are more about personal and psychological sensations when wearing the device, while the remaining three categories focus on the device's body feel. The participants are able to choose a value between 0 and 10 for every category. 0 means it has low impact, and 10 a high impact.

Title	Description
Emotion	I am worried about how I look when I wear this device. I feel tense or on edge because I am wearing the device.
Attachment	I can feel the device on my body. I can feel the device moving.
Harm	The device is causing me some harm. The device is painful to wear.
Perceived change	Wearing the device makes me feel physically different. I feel strange wearing the device.
Movement	The device affects the way I move. The device inhibits or restricts my movement.
Anxiety	I do not feel secure wearing the device.

Table 1: Comfort Rating Scale (CRS) categories as proposed in [12]. The CRS includes 6 categories: Emotion, Attachment, Harm, Perceived change, Movement and Anxiety.

Data Preparation

For evaluation purposes a ground truth is needed. Therefore we used a synchronization gesture. In our case the synchronization gesture has the requirement that both hand a head needs to follow the same movement. Therefore a simple vertical jump was chosen. This creates a clearly identifiable peak in the acceleration data. The gesture was done at the beginning and at the end of the recording and thus marks both in the data.

To be able to synchronize two independent recorded signals we need to preprocess the data. The first step in the data preparation process was to crop out the data at these marks. Due to sample losses, for example through the the wireless connection (BT or WiFi) or not fully achieved sampling rates, the signals are initially of different length. Furthermore they need to be sampled with the same sampling rates. Therefore the Platypus data must be sampled down to 50 Hz. Since we are only able to set a label for a certain sequence using the eSense, we have to set the timestamps of the eSense as ground truth for all other body sensors involved in the synchronization process. Under these circumstances we are now able to calculate all sensor signals equidistantly. Due to simulate the case that no synchronization gesture is available

to synchronize the data, the time series of the eSense was shortened by 10% at the beginning and at the end.

Data Synchronization Method

Since the method for synchronizing signals is of central importance, it takes up most of the work presented here. The following table describes step wise the developed algorithm and the results after every step. When the algorithm finishes we are able to propagate the label throughout the sensors. Parameters like the window-size and window-length, but also the threshold of the binary filter, can be adjusted variably. In the first version of the algorithm a simple ASCII character is written with a button press at the beginning and end of the activity. In the future we plan to use the microphone and voice-to-speech recognition for setting the label.

Step	Name	Description
1	Dimension reduction of raw data	Calculate vector length per sample (dimension reduction from 3D to 1D).
		Result: Signals with reduced dimension.
2	Windowing	Data is divided into windows. Window length and overlap ratio can be set variable.
		Result: Windowed data.
3	Feature calculation	Calculation of the standard deviation.
		Result: Standard deviation per window.
4	Binary Filter	Both standard deviation signals are passed through a binary filter. A threshold is used to decide whether it's a sedentary or a non-sedentary activity. 0 (sedentary) if the current value is smaller than threshold and 1 (non-sedentary) if higher than the threshold.
		Result: Two signals with the values 0 and 1. 0 for sedentary sequences and 1 or non-sedentary activities.
5	Cross-Correlation	Cross-correlation [3] of both binary filtered signals. The eSense signal is cross correlated with the other signals.
		Result: Cross-Correlation coefficiants.
6	Index Selection	The window with the highest correlation coefficient marks the best index synchronize the signal.
		Result: Start window for synchronizing
7	In-Window-Cross-Correlation	Cross correlation for all samples in this window.
		Result: Exact index for synchronization.
8	Label propagation	Labels from the eSense signal can be copied to the other sensor data.
		Result: Labeled data.

Table 2: Step by step explanation of the algorithm. The algorithm is divided into 8 steps. First the dimension of the data is reduced by calculating the vector length, divided into windows and finally the standard deviation is calculated. The standard deviations are now passed through a binary filter, which writes a 0 for sedentary activity and a 1 for non-sedentary activity. Both signals are then cross-correlated. The position of the highest correlation can then be used to deduce the synchronization point in the initial signal.

5 RESULTS AND DISCUSSION

Comfort Rating Scale

Figure 2 shows the result of our CRS study. To sum the result up we can say that in general the device is comfortable

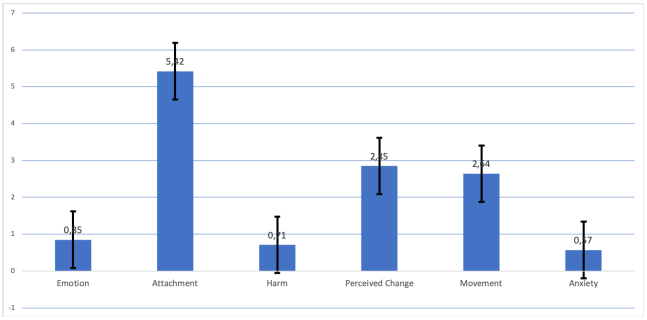


Figure 2: CRS result means and standard deviation. Anxiety: 0.57, 0.6; Movement: 2.64, 3.09; Perceived Change: 2.85, 3.17; Harm: 0.71, 0.8; Attachment: 5.42, 2.38; Emotion: 0.85, 1.31.

to wear, but sometimes you can feel it moving in your ear. One participant in the study noted that the earplugs tend to fall out of the ear during heavy movement, like dribbling a basketball, even if adjusted correctly. The average values for the Emotion, Harm and Anxiety categories show that users are generally not concerned about their appearance. This is certainly due to the fact that earbuds are very inconspicuous and devices like these have long since found their way into our everyday lives. The results in the other categories vary. This shows the standard deviation. The perceived wearing comfort is strongly user-dependent and is probably also related to the individual shape of the inner ear. This is as unique as the fingerprint [4], which is why it is difficult to develop a shape that everyone feels comfortable with.

Data Synchronization Method

In order to investigate the reliability in terms of automatically synchronizing the inertial data streams, we decided to use the time and sample mismatch between the ground truth and the index used as the synchronization point. To evaluate the performance of our algorithm we first calculated the best working parameters for window size and overlap ratio by using a brute force method. This was possible because of the short computational time and, compared to long-term benchmark data, limited amount of data. The determined parameters from these experiments were found to be:

- **window-size:** 50 samples
- **overlap-ratio:** 85%

With these parameters fixed, we calculated the time mismatch separately for every inertial data recording, as depicted in Table 3. The graphical representation as given out by our algorithm are shown in the Figures 3, 5 and 4.

These figures present two different signals: The top one is the one recorded at the wrist by the Platypus prototype. The bottom plot contains the inertial data from the eSense. The ground truth, as obtained from synchronization gestures

before and after the recording (not shown), as a reference is plotted transparently. Overlaying the ground truth the resulting shortened and synchronized signal is depicted, with the vertical red lines marking the beginning and ending of the calculated synchronization point. The black line-plot embedded in each bottom plots shows the correlation signal. The inertial signals are synchronized according to the highest cross-correlation.

Record	Mismatch in Samples	Mismatch in Seconds	Activity
1	15	0.30	1, 3, 5, 6
2	16	0.32	1, 6
3	16	0.32	1, 4, 6
4	20	0.40	1, 6
5	21	0.42	1, 2, 5, 6
6	21	0.42	1, 4, 6
7	23	0.46	1, 3, 5, 6
8	87	1.47	1, 3, 5, 6
9	293	5.86	1, 3, 5, 6
10	386	7,72	1, 4
11	418	8.36	5
12	874	17.48	1, 4
13	1195	23.90	1, 2, 3, 5
14	1742	34.84	1, 4

Table 3: Synchronization error per record in samples and seconds. The records are recordings from 7 participants and overall 6 different activities. Synchronization tends to be within one second for records that contain clear sequences of activities that contain different intensities. (1) read or desk work, (2) walk, (3) climbing stairs, (4) sit, (5) dribble a basketball and (6) pause or rest phase.

In the first version of the algorithm the binary filter was not yet part of it, which resulted in problems synchronizing correctly, if no pause phases has been part of the record, for example record 11 in Table 3. The binary filter sets a very hard boundary between sedentary and non-sedentary activities, decided by a threshold, wherefore we needed to have a closer look on the calculated standard deviation signal. Here we saw that the threshold needs to be between 0.500 mg and 0.515 mg. After setting the boundaries we evaluated that the best working threshold is at 0.508 mg. The mismatch (median) of the algorithm was 61 samples or 1.22 seconds. Due to the usage of the binary filter we were able to improve our results to 55 samples or 1.10 seconds of mismatch. The records that could be rather poorly synchronized with our algorithm are records that mostly consists of sedentary activities as e.g. sitting, reading or desk work, as depicted in Figure 5 or records with heavy movements, but without pause phases, fig. 4. Very well devoted, data sets can be synchronized that reflect activities involving a high degree of locomotion as well as sufficient phase of pauses, e.g. figure 3.

- **Minimum time mismatch:** 0.30 seconds or 15 samples

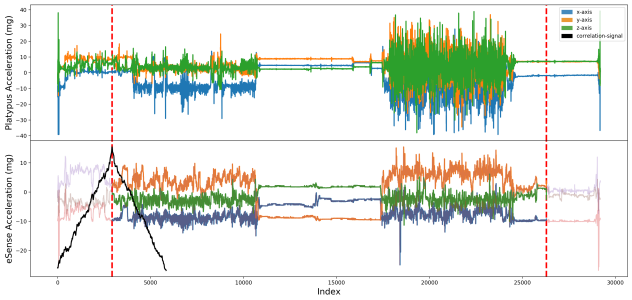


Figure 3: Best synchronization with a mismatch of 0.30 seconds. The figure shows that the synchronization works best with sufficient long periods of sedentary activity. The inertial signal of the wrist (top) compared with the synchronized signals of the head (bottom). The black signal at the bottom left depicts the cross-correlation between the binary filtered signals.

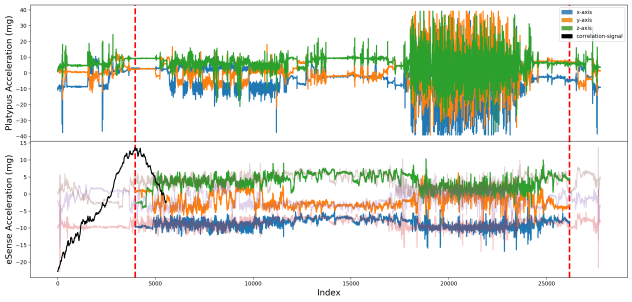


Figure 4: Data without sufficient pause phases, with plots defined as in Figure 3, using Record 13 in table 3. Our algorithm's synchronization was off by 1195 samples or 23.90 seconds.

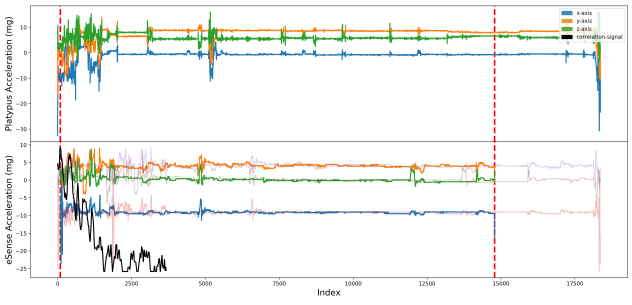


Figure 5: Worst-case synchronization with a mismatch of 34.84 seconds, record 14 in table 3. In this data almost only desk work has been performed. The inertial signal of the wrist (top) compared with the synchronized signals of the head (bottom). The black signal at the bottom left depicts the cross-correlation between the binary filtered signals.

- **Maximum time mismatch:** 34.84 seconds or 1742 samples
- **Median time mismatch:** 1.10 seconds or 55 samples

6 CONCLUSIONS

We presented in this paper a novel annotation method for recording activity recognition benchmark data. Our method relies on users wearing a small earbud-like device in their ear, which is equipped with a button and an inertial measurement unit. The inertial data from the ear-worn sensor are synchronized to all other data via cross-correlation, after which the user-presses serve as labels that annotate all sensor streams. In a preliminary study with 7 users, we investigated how well this synchronization works, as well as how comfortable the earbud-like wearable was to our study volunteers.

This paper offers a first approach to spread the annotations temporally correct over any number of sensors and to synchronize time series that have been recorded at the same time from different devices. If the data contains sequences that can be uniquely assigned to an activity, with sufficient periods of resting activity, the synchronization was found to be sufficiently reliable. However, the algorithm does not work reliably enough if the head and hand movements during an activity do not basically follow the same direction or if they can completely differ from each other. In addition, care must be taken to ensure that the movements follow a pattern that includes rest periods. The evaluation, as in Table 3, has shown that these are essential for reliable synchronization.

In terms of wearing comfort, we found that the used eSense prototype is highly promising as an annotation tool for everyday recordings 'in the wild'. The fact that it can be worn comfortably, with attachment as a weakest link for some participants, and almost hidden in the ear makes it ideal for recording *and* annotating data outside our laboratory. As such devices could be operated simultaneously as wireless headsets, the one remaining hurdle for use of our method in long-term and day-long activity recordings is the eSense's battery.

ACKNOWLEDGMENTS

We would like to thank all study participants, funding was supplied by the FFG's MinIAttention Project (BRIDGE Fruehphase, 3. Ausschreibung 2014, Projekt 851227).

REFERENCES

[1] Manfred Benesch, Helmuth Kubin, and Klaus Kabitzsch. 2017. Processing of Real-time Data in Big Manufacturing Systems.

[2] Ulf Blanke, Diane Larlus, Kristof Van Laerhoven, and Bernt Schiele. 2010. Standing on the Shoulders of Other Researchers A Position Statement. In *Pervasive Workshop*.

[3] The SciPy community. 2019. Numpy Documentation For Cross-Correlation. https://docs.scipy.org/doc/numpy/reference/generated/numpy.correlate.html. Last updated on July 26, 2019, Accessed: 2019-08-11.

[4] A. H. Cummings, M. S. Nixon, and J. N. Carter. 2010. A novel ray analogy for enrolment of ear biometrics. In *2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*. 1–6.

[5] Brian Derek DeBusschere. U.S. Patent 2016/0338599 A1, Nov. 24, 2016. Synchronizing Cardiovascular Sensors for Cardiovascular Monitoring.

[6] Daniel Roggen et al. 2010. Collecting complex activity datasets in highly rich networked sensor environments. *2010 Seventh International Conference on Networked Sensing Systems (INSS)* (2010), 233–240.

[7] Joel Hesch et al. U.S. Patent 2015/0185054A1, Jul. 2, 2015. Methods and Systems for Synchronizing Data Received From Multiple Sensors of a Device.

[8] Neal Crook et al. U.S. Patent 8,711,238 B2, Apr. 29, 2014. Systems and Methods for Synchronizing and Controlling Multiple Image Sensors.

[9] Radu Pitigoi-Aron et al. U.S. Patent 9,436,214 B2, Sep. 6, 2016. System And Methods of Reducing Energy Consumption by Synchronizing Sensors.

[10] Alexander Hölzemann and Kristof Van Laerhoven. 2018. Using Wrist-Worn Activity Recognition for Basketball Game Analysis. In *Proceedings of the 5th international Workshop on Sensor-based Activity Recognition and Interaction - iWOAR '18*. ACM Press.

[11] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (jul 2018), 83–89.

[12] James F. Knight, Chris Baber, Anthony Schwirtz, and Huw William Bristow. 2002. The comfort assessment of wearable computers. *Proceedings. Sixth International Symposium on Wearable Computers*, (2002), 65–72.

[13] Kai Kunze, Paul Lukowicz, Holger Junker, and Gerhard Tröster. 2005. Where am I: Recognizing On-body Positions of Wearable Sensors. In *LoCA*.

[14] Christoph Lenzen, Philipp Sommer, and Roger Wattenhofer. 2015. PulseSync: An Efficient and Scalable Clock Synchronization Protocol. *IEEE/ACM Transactions on Networking* 23, 3 (Jun 2015), 717–727.

[15] Lin Lin, Chengfeng Yang, Maode Ma, Shiwei Ma, and Hao Yan. 2016. A clock synchronization method for molecular nanomachines in bio-nanosensor networks. *IEEE Sensors Journal* 16, 19 (2016), 7194–7203.

[16] Aneeq Mahmood, Reinhard Exel, Henning Trsek, and Thilo Sauter. 2016. Clock synchronization over IEEE 802.11 - A survey of methodologies and protocols. *IEEE Transactions on Industrial Informatics* 13, 2 (2016), 907–922.

[17] Tal Mizrahi. 2017. Clock synchronization using multiple network paths. US Patent 9,806,835.

[18] Christoph Sens-Schönfelder. 2008. Synchronizing seismic networks with ambient noise. *Geophysical Journal International* 174, 3 (Sep 2008), 966–970.

[19] Joshua Abraham Tabak. U.S. Patent 9,646,650 B2, May 9, 2017. Automatically Syncing Recordings Between Two Or More Content Recording Devies.

[20] Liyue Zhao, Gita Reese Sukthankar, and Rahul Sukthankar. 2011. Robust Active Learning Using Crowdsourced Annotations for Activity Recognition. In *Human Computation*.

The CAMS eSense Framework – Enabling Earable Computing for mHealth Apps and Digital Phenotyping

Jakob E. Bardram
jakba@dtu.dk
Copenhagen Center for Health Technology
Department of Health Technology, Technical University of Denmark
Copenhagen, Denmark

ABSTRACT

Earable computing devices can be an important platform for mobile health (mHealth) applications and digital phenotyping, since they allow for collection of detailed sensory data while also providing a platform for contextual delivery of interventions. In this paper we describe how the eSense earable computing platform has been integrated with a programming framework and runtime platform for the design of mHealth applications. The paper details how this programming framework can be used in the design of custom mHealth technologies. It also provide data and insight from an initial study in which this framework was used to collect real-life contextual data, including sensory data from the eSense device.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; • **Software and its engineering** → **Development frameworks and environments**; • **Applied computing** → Health informatics.

KEYWORDS

digital phenotyping, mobile health, mobile sensing, eSense, earable computing

ACM Reference Format:
Jakob E. Bardram. 2019. The CAMS eSense Framework – Enabling Earable Computing for mHealth Apps and Digital Phenotyping. In *Proceedings of EarComp 2019: 1st International Workshop on Earable Computing (EarComp 2019).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

A significant body of research has been applying mobile sensing to health and wellness applications (mHealth technologies) [3] including, for example, the EmotionSense [11], BeWell [10], and StudentLife [15] which are systems that classify physical activity, sleep, and social interaction based on sensor data. Similarly, studies in mental health have demonstrated correlations and predictive power between phone-based features on physical activity, mobility, social activity, phone usage, and voice data on the one side, and mental health symptoms in e.g., depression [14], bipolar disorder [5, 7], and schizophrenia [4] on the other side.

More generally, it has been argued that obtaining a more precise understanding of a disease can happen in multiple dimensions, and one new dimension is the use of mobile devices to measure people's activity and other factors more continuously and accurately. As such, mobile sensing has been defined as central to the 'Precision Medicine' initiative; genotypic information is only powerful if phenotypic information is also available [1]. The use of everyday mobile and wearable technology for collection of behavioral, psychological, and health data has been termed 'digital phenotyping' [8, 13], which can be defined as;

> *continuous and unobtrusive measurement and inference of health, behavior, and other parameters from wearable and mobile technology.*

Earable computing devices provide a novel and significant technological platform for the design of mobile health (mHealth) technologies and digital phenotyping. First of all because earable computing enables new sensor modalities and the collection of a new type of data, including head movement (from accelerometers and gyroscopes), sound and noise levels as experienced by the user (and not by the phone which might be in a pocket), as well as more health and well-being features related to e.g. cardio-vascular activity (pulse, heart rate (HR), and heart rate variability (HRV)), sleep detection, etc. But secondly also because earable computing might be a platform for delivering Just-in-Time Adaptive Interventions (JITAI) [12], by coupling contextual sensing with the delivery of a personalized and private intervention using the headset speakers. Hence, an mHealth intervention no longer needs to rely on notifications on the user's phone screen (with all the problems of notification fatigue associated with this), but instead can be delivered as small audio messages targeted for the specific person only. Thirdly, if the earable device has some input modalities – like a push button – simple Ecological Momentary Assessment (EMA) sampling can be done.

The eSense device from Nokia Bell Labs provide such an earable computing platform [9]. In combination with a more general data sampling platform, this device can provide important additional sensing modalities for digital phenotyping as well as in the design of mHealth applications. In this paper we describe how the eSense technology has been integrated into a larger runtime platform and programming framework for digital phenotyping and mHealth application development, which then allows researchers and mHealth application designers to achieve the visions for earable computing, as outlined above. The paper also reports from a small study in

EarComp 2019, September 10, 2019, London, UK

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

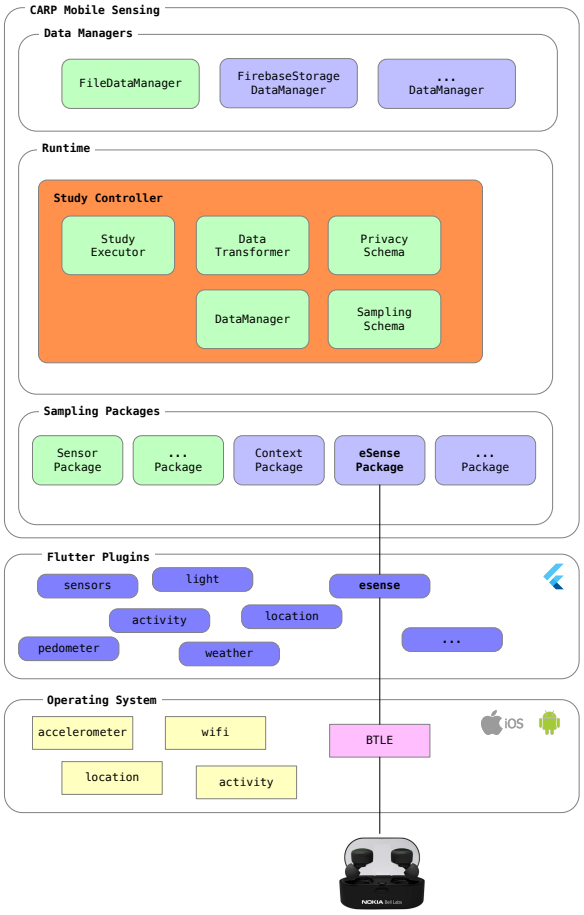


Figure 1: A simplified view on the CAMS architecture highlighting how the eSense device is integrated into the framework using both a Flutter plugin and a CAMS sampling package.

which the eSense device was used in data collection and where earable sensor data is combined with 'traditional' mobile sensing modalities, like location, connectivity, communication patterns, activity recognition, noise, etc. The earable computing programming framework presented in this paper is open source and we hope that others in the earable community can benefit from using this.

2 CAMS ESENSE FRAMEWORK

The eSense technology has been integrated into the CARP Mobile Sensing (CAMS) framework [2][1]. CAMS have been described elsewhere and there are plenty of online resources in terms of application programming interface (API) documentation and tutorial available – see Appendix A.2. CAMS is a cross-platform (iOS & Android) programming framework for building mHealth technology that incorporates mobile and wearable sensing. It is designed to be highly extensible allowing for; (i) adding new data sampling modalities (such as wearable devices like the eSense device), (ii) implementing different kinds of data transformation (e.g. transforming sensor data to standardized formats or on-phone pre-processing before upload), (iii) using data sampling actively in app design, and (iii) supporting different data off-loading strategies (such as local file storage or cloud-based upload of data). CAMS is implemented in Flutter, which is Google's cross-platform portable toolkit for building natively-compiled applications for mobile, web, and desktop from a single codebase [6]. Flutter rely on Dart, which is a modern object-oriented, reactive programming language optimized for non-blocking user-interface programming with a mature and complete async-await event-driven code style, paired with isolate-based concurrency.

As illustrated in Figure 1, CAMS has three main layers; (i) a runtime layer (in the middle), (ii) a set of data mangers (top layer), and (iii) a set of sampling packages (bottom layer). CAMS is a very flexible and extensible 'plug-and-play' architecture, in which most of the components show in Figure 1 can be tailored, extended, or replaced, and customized components can be added. For example, a new sampling package can be added, which supports sampling of data from a new source – both from on-board phone sources (such as a phone sensor or log) or off-board wearable sensors which can be accessed e.g. via Bluetooth Low Energy (BTLE). The integration of eSense into CAMS is an example of the latter, where an eSense sampling package have been implemented, which then can be linked and used in app development. The implementation of a sampling package in CAMS rely on access one or more Flutter plugins for data access. These Flutter plugins are strictly speaking not a part of CAMS, but is a generic way to access the phone's operating system (OS) in a cross-platform manner. A Flutter plugin is often implemented using the 'Platform Channel' technology in Dart/Flutter, which allow Flutter to access the native OS API on both Android and iOS[2].

Hence, in order to support the eSense device in Flutter, we have implemented two components; (i) a Flutter plugin which uses a platform channel to access the eSense Java API, and (ii) an eSense sampling package which integrates support for eSense into CAMS.

2.1 The eSense Flutter Plugin

The eSense Flutter plugin has been designed to resemble the Android eSense API almost 1:1 and the eSense Android programmer will be able to recognize the names of the different classes, methods, and class variables. For example, the methods on the `ESenseManager` class is mapped 1:1. However, one major design change has been done; the eSense Flutter plugin follows the Dart/Flutter reactive programming architecture using *streams*. Hence, you do not 'add listeners' to an eSense device (as you do in Java) – rather, you obtain a Dart stream and listen to this stream, and utilize all the other very nice stream operations that are available in Dart – including creating very beautiful reactive user interfaces (UIs). Listing 1 shows the basic Dart code on how to use the eSense plugin. As can be seen, it quite straight-forward and it only requires a few lines of code to use the plugin. The reader familiar with the eSense Java API will recognize the way to use the API and its names.

[1]CARP is an abbreviation of the CACHET Research Platform.

[2]More information on how to write platform-specific code in Flutter is available at https://flutter.dev/docs/development/platform-integration/platform-channels.

```dart
import 'package:esense/esense.dart';

// listen to connection events before connecting
ESenseManager.connectionEvents.listen((event)
    => print('CONNECTION event: $event');

// try to connect to the eSense device with a given name
success = await ESenseManager.connect('eSense-0332');

// listen to sensor events and print them
ESenseManager.sensorEvents.listen((event)
  => print('SENSOR event: $event');
```

Listing 1: Using the eSense Flutter plugin.

Note that playing and recording audio are performed via the Bluetooth Classic interface and are not supported by the eSense plugin. However, as we shall present below, CAMS supports sampling of audio and noise, which is done via the eSense device microphone, once connected.

The eSense Flutter plugin has been released to the Flutter package sharing site[3] with documentation on the API and how to use the plugin[4]. Links to online resources are provided in Appendix A.1. At the time of writing, the eSense Flutter plugin only works on Android, since Nokia Bell Labs initially only provided an Android Java API. However, a 3rd party iOS API has just been released and support for iOS can be implemented in the eSense Flutter plugin. This means that all eSense apps implemented using Flutter and CAMS would run on both Android and iOS without *any* platform-specific development needed. This is the true strength of using Flutter.

2.2 The eSense Sampling Package

A CAMS sampling package basically consists of three components; (i) a `Measure` which defines *what data to collect*, (ii) a `Probe` that implements *how data is collected*, and (iii) a `Datum` object which specify the *data format* of the collected data. Two types of data can be collected from the eSense device; (a) button pressed / released events and (b) sensor events from the device's inertial measurement unit (IMU) (accelerometer and gyroscope). CAMS supports both these type of measures, using two different probes, and stores it in two different datum objects. Hence, the two types of measures are independent and the app developer can choose to use one and/or the other.

Listing 2 shows how the eSense measures are configured to be part of a CAMS study. A `Study` object is created with a name and a file storage as the data endpoint (line 1–3), and then a list of measures are added to a task and a trigger, which basically just starts the sampling immediately and runs forever (line 4–15). The study is configured to sample eSense button events, eSense sensor events, noise, location, activities, local weather information, and scans for Bluetooth devices in the phone's proximity. The exact configuration of these measures is defined in a so-called `SamplingSchema`, where the `common` schema is used in this case.

```
study = Study('1234', 'user@dtu.dk')
  ..name = 'CARP Mobile Sensing - eSense sampling demo'
  ..dataEndPoint = getDataEndpoint(DataEndPointTypes.FILE)
  ..addTriggerTask(
    ImmediateTrigger(),
    Task()
      ..measures = SamplingSchema.common().getMeasureList([
        ESenseSamplingPackage.ESENSE_BUTTON,
        ESenseSamplingPackage.ESENSE_SENSOR,
        AudioSamplingPackage.NOISE,
        ContextSamplingPackage.LOCATION,
        ContextSamplingPackage.ACTIVITY,
        ContextSamplingPackage.WEATHER,
        ConnectivitySamplingPackage.BLUETOOTH,
      ]));
```

Listing 2: Using the eSense measure types as part of a CAMS study.

[3]http://pub.dev
[4]https://pub.dev/packages/esense

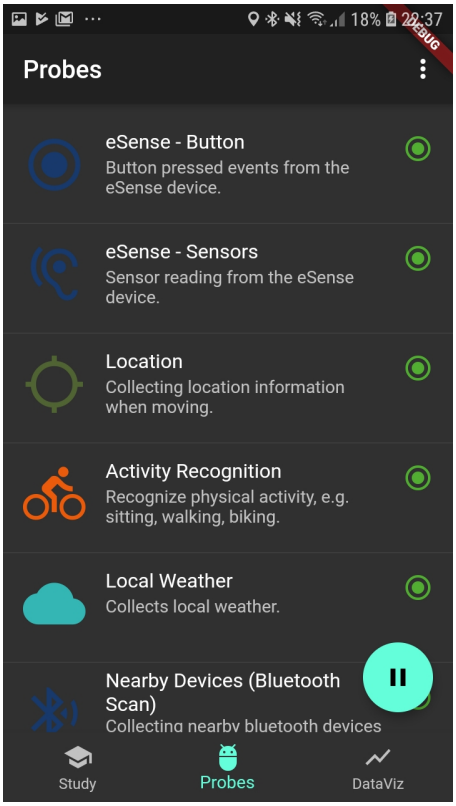


Figure 2: The study configured in Listing 2 shown in the list of probes in the CAMS mobile sensing app.

Once the study is defined, it can be handed over to the CAMS `StudyController` as shown in Listing 3. This paper do not allow for going into the details of CAMS, but the code examples hopefully illustrates that a sampling study can be configured and executed quite easily.

```dart
// Create a controller for this study, initialize it, and start it
controller = StudyController(study);
await controller.initialize();
controller.start();

// listening on all data events from the study and print it
controller.events.forEach(print);
```

Listing 3: Starting a study.

3 ESENSE STUDY

In order to evaluate the eSense Flutter plugin and the eSense CAMS sampling package, we created the study listed in Listing 2 and deployed it in the CAMS client app. This study configuration invokes a set of corresponding probes, which samples the specified data types. The CAMS client app with this list of probes is shown in Figure 2. The following measures were configured:

- eSense button events
- eSense sensor events with a sampling rate of 10 Hz
- Ambient noise, sampled over a 5 seconds window every 45 seconds.
- Location triggered by the phone on movement.
- Activity as recognized by the activity recognition API on the phone.
- Local weather as collected from the WeatherAPI service.
- Scanning of nearby Bluetooth devices every 60 seconds.

The main goal of this sampling schema is to sample data in a scenario where a user is physically active during a day, performing different activities (e.g. biking, walking, sitting), at different locations, with different weather, with different noise levels, and with different people (the Bluetooth scan). The eSense sensor data is collected to see if this can be correlated or used in classification of activities. The user is instructed to press the eSense button when starting a new activity and/or changing context.

We ran the study for one day which included activities of walking, driving, sitting at a desk, and biking. During the study we collected more than 150,000 data points (61 MB data), of which the vast majority was the detailed sensor data from the eSense IMU sensor. All of this data was stored locally on the phone (hence, offloading to e.g. Firebase was not used in this study). The app and the eSense device ran continuously during the entire day (8 hours) without any significant breakdowns and problems. However, valuable experience in handling the eSense device was obtained. For example, there might be inference between the eSense Bluetooth connectivity while doing the Bluetooth scan in CAMS and the BTLE connection to the eSense IMU could break, if the earplug was used for streaming music. Hence, this study gave some input in how to make the data sampling package and the eSense probes more robust in handling disconnection and re-connection scenarios. At the time of writing the collected data has not been analysed. But the study demonstrated the feasibility of using CAMS with the eSense sampling package in such digital phenotyping studies. The data is available for download – see Appendix A.3 for details.

4 CONCLUSION

This paper has presented the integration of the eSense earable computing platform from Nokia Bell Labs into the CARP Mobile Sensing (CAMS) framework. This integration allows software developers of mHealth apps to include mobile and wearable sensing to their app design, which now also include the eSense device. Moreover, the framework allows for using eSense in digital phenotyping, as demonstrated by a small study. Now that this infrastructure is in place, we plan to set up more studies where the eSense earable computing technology can be used in combination with all the other sampling measures available in CAMS.

ACKNOWLEDGMENTS

This work has been funded by the Copenhagen Center for Health Technology (CACHET) [www.cachet.dk].

REFERENCES

[1] Euan A. Ashley. 2015. The precision medicine initiative: A new national effort. *JAMA* 313, 21 (2015), 2119–2120.
[2] Jakob E. Bardram. 2019. The CARP Mobile Sensing Framework: A Cross-platform, Reactive, Programming Framework and Micro-service Runtime Environment for Digital Phenotyping. *In Submission* (2019).
[3] Jakob E Bardram and Mads Frost. 2016. The Personal Health Technology Design Space. *IEEE Pervasive Computing* 15, 2 (2016), 70–78.
[4] Ian Barnett, John Torous, Patrick Staples, Luis Sandoval, Matcheri Keshavan, and Jukka-Pekka Onnela. 2018. Relapse prediction in schizophrenia through digital phenotyping: a pilot study. *Neuropsychopharmacology* (2018), 1.
[5] Maria Faurholt-Jepsen, Jonas Busk, Mads Frost, Maj Vinberg, Ellen M Christensen, Ole Winther, Jakob E Bardram, and Lars V Kessing. 2016. Voice analysis as an objective state marker in bipolar disorder. *Translational psychiatry* 6, 7 (2016), e856.
[6] Flutter 2019. Flutter – Google's portable UI toolkit for building beautiful, natively-compiled applications for mobile, web, and desktop from a single codebase. Retrieved July 12, 2019 from https://flutter.dev
[7] Agnes Grünerbl, Amir Muaremi, Venet Osmani, Gernot Bahle, Stefan Oehler, Gerhard Tröster, Oscar Mayora, Christian Haring, and Paul Lukowicz. 2014. Smartphone-based recognition of states and state changes in bipolar disorder patients. *IEEE Journal of Biomedical and Health Informatics* 19, 1 (2014), 140–148.
[8] Sachin H Jain, Brian W Powers, Jared B Hawkins, and John S Brownstein. 2015. The digital phenotype. *Nat Biotech* 33, 5 (may 2015), 462–463.
[9] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (2018), 83–89.
[10] Nicholas D Lane, Mashfiqui Mohammod, Mu Lin, Xiaochao Yang, Hong Lu, Shahid Ali, Afsaneh Doryab, Ethan Berke, Tanzeem Choudhury, and Andrew Campbell. 2011. Bewell: A smartphone application to monitor, model and promote wellbeing. In *5th international ICST conference on pervasive computing technologies for healthcare*. 23–26.
[11] Neal Lathia, Veljko Pejovic, Kiran K Rachuri, Cecilia Mascolo, Mirco Musolesi, and Peter J Rentfrow. 2013. Smartphones for Large-Scale Behavior Change Interventions. *IEEE Pervasive Computing* 12, 3 (2013), 66–73.
[12] Inbal Nahum-Shani, Shawna N Smith, Bonnie J Spring, Linda M Collins, Katie Witkiewitz, Ambuj Tewari, and Susan A Murphy. 2017. Just-in-Time Adaptive Interventions (JITAIs) in Mobile Health: Key Components and Design Principles for Ongoing Health Behavior Support. *Annals of Behavioral Medicine* 52, 6 (12 2017), 446–462.
[13] Jukka-Pekka Onnela and Scott L Rauch. 2016. Harnessing Smartphone-Based Digital Phenotyping to Enhance Behavioral and Mental Health. *Neuropsychopharmacology* 41, 7 (2016).
[14] Sohrab Saeb, Emily G Lattie, Stephen M Schueller, Konrad P Kording, and David C Mohr. 2016. The relationship between mobile phone location sensor data and depressive symptom severity. *PeerJ* 4 (2016), e2537.
[15] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. 2014. StudentLife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 3–14.

A ONLINE RESOURCES

This appendix provides link to different online resources relevant for the eSense Flutter plugin, CAMS, and the CAMS eSense sampling packages, as well as the data from the small study reported in this paper.

A.1 eSense Flutter Plugin

- The `esense` Plugin at `pub.dev` – https://pub.dev/packages/esense
- The `esense` API documentation – https://pub.dev/documentation/esense/latest/
- The `esense` Plugin GitHub – https://github.com/cph-cachet/flutter-plugins/tree/master/packages/esense

A.2 CAMS Framework and Documentation

- The CARP Mobile Sensing (CAMS) core Flutter Plugin at `pub.dev` – https://pub.dev/packages/carp_mobile_sensing
- The CAMS tutorials and documentation – https://github.com/cph-cachet/carp.sensing-flutter/wiki
- The CAMS API documentation – https://pub.dev/documentation/carp_mobile_sensing/latest/
- The CAMS eSense Sampling Package at `pub.dev` – https://pub.dev/packages/carp_esense_package
- The CAMS GitHub – https://github.com/cph-cachet/carp.sensing-flutter

A.3 Data from the eSense Study

The data and description from this small (N=1) study can be accessed from:

- https://github.com/cph-cachet/data/tree/master/2019.08.28.eSense

Using the eSense Wearable Earbud as a Light-Weight Robot Arm Controller

Henry Odoemelem
University of Siegen
henry.odoemelem@student.uni-siegen.de

Alexander Hölzemann
University of Siegen
alexander.hoelzemann@uni-siegen.de

Kristof Van Laerhoven
University of Siegen
kvl@eti.uni-siegen.de

ABSTRACT

Head motion-based interfaces for controlling robot arms in real time have been presented in both medical-oriented research as well as human-robot interaction. We present an especially minimal and low-cost solution that uses the eSense [1] ear-worn prototype as a small head-worn controller, enabling direct control of an inexpensive robot arm in the environment. We report on the hardware and software setup, as well as the experiment design and early results.

CCS CONCEPTS

• **Computer systems organization** → *Embedded systems*; **Robotics**; • **Human-centered computing** → *Ubiquitous and mobile computing*.

ACM Reference Format:
Henry Odoemelem, Alexander Hölzemann, and Kristof Van Laerhoven. 2019. Using the eSense Wearable Earbud as a Light-Weight Robot Arm Controller. In *EarComp 2019*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/0000000.000000000

1 INTRODUCTION

The past decade has seen an increasing interest in developing and investigating the interface between humans and robots. For many scenarios in this research, precision and speed are crucial, with cost factors and size and mobility of such systems being less of a focus. In contrast, we present here a human-robot interface that aims at being minimal in terms of size and costs, and intend to investigate the trade-offs that are caused by this minimalism in terms of accuracy and speed. Our application domain is head motion-based robot control, as it is for instance required to enable tetraplegics to control a multi-degree of freedom robot arm in real-time using solely head motion (as for instance motivated in [4]).

EarComp 2019, September 10, 2019, London, UK

ACM ISBN 978-1-4503-6317-4/19/07.
https://doi.org/10.1145/0000000.000000000

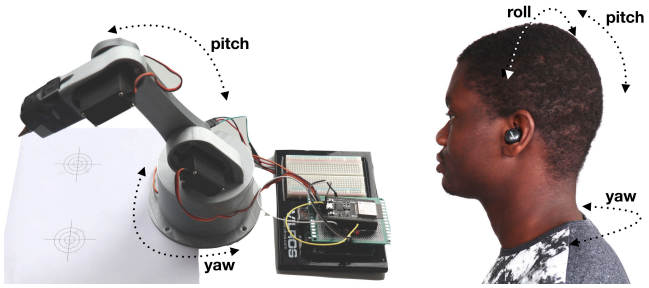


Figure 1: This work presents an affordable and light-weight system that uses an in-ear IMU to control a robot arm's yaw and pitch motions by moving the head.

2 DESIGN OF HARDWARE AND SOFTWARE

For our design, we use an open-source design 3-DOF robot arm, using STL Files available on [1]. This design uses 3 low-cost mg996r servo motors capable of 180 deg rotation and is powered by an Arduino (connected to a 9V dc power adapter). This design results in an extensible, re-programmable, and low-cost robot arm that costs about 70$ for all components. We use this model as this design explicitly represents an entry model for interactive control applications, where users need to adapt to the speed and accuracy of the servo motors.

In order to control the robot arm, IMU (Inertial Measurement Unit) data are sent via Bluetooth Low Energy(BLE) from an eSense ear-worn unit to a ESP32-WROOM-32D-equipped robot for local processing. eSense[2] is a multi-sensory earable platform for personal-scale behavioural analytics research. It is a True Wireless Stereo (TWS) earbud, composed of a Qualcomm CSR8670, with dual mode Bluetooth (Bluetooth Classic and Bluetooth Low Energy), three-axis accelerometer, a three-axis gyroscope,etc. The left earbud is the one containing the IMU sensor accessible through the BLE interface. The ESP32-WROOM-32D module[3] at the robot arm. This module is a generic Wi-Fi+BT+BLE MCU system-on-chip, equipped with an Xtensa single-/dual-core 32-bit LX6 microprocessor. With up to 600 MIPS, it is designed for flexible mobile, wearable, and networked sensor applications.

[1] https://howtomechatronics.com/download/arduino-robot-arm-stl-files/
[2] eSense User Documentation: http://www.esense.io/share/eSense-User-Documentation.pdf
[3] ESP32 Datasheet: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32d_esp32-wroom-32u_datasheet_en.pdf

Connection: The ESP32 microcontroller is programmed using the Arduino IDE, and the BLE connection between eSense and ESP32 was accomplished with the help of the Arduino BLEDevice library [4]. To access the IMU data using the appropriate service and characteristics UUID[5], we first have to register for notification and enable IMU sampling, here we are using 50Hz sampling rate. Next, on each notification we convert the accelerometer readings to g using the 8192 LSB/g Scale factor (+/- 4 g range is used) and gyroscope reading to deg/s using 65.5 LSB/(deg/s) Scale factor(+/-500 deg/s range is used).

Frame transformation: The acceleration in the earth's reference frame has to be transformed to the IMU body frame, the orientation of the eSense IMU is shown in the figure 3. Appendix A contains the details on how we achieve this.

When the eSense is worn, the orientation of the earbud is as shown in the figure 4, where it is 90 deg (approximately) rotated about the z-axis from the orientation we have used in our calculations, to account for this and be able to make use of (14), we must first make a transformation of our IMU readings from the worn orientation to the original orientation used in calculation as shown in (1), where variables with subscript w represent values from the worn orientation.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos(-90) & -\sin(-90) & 0 \\ \sin(-90) & \cos(-90) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = \begin{pmatrix} y_w \\ -x_w \\ z_w \end{pmatrix} \quad (1)$$

The simple substitution necessary is as shown in (2) and (3).

$$G_x = G_{yw}, G_y = -G_{xw}, G_z = G_{zw} \quad (2)$$

$$gyro_x = gyro_{yw}, gyro_y = -gyro_{xw}, gyro_z = gyro_{zw} \quad (3)$$

Filtering: Accelerometer is susceptible to high-frequency noise and needs a low pass filter, while the gyroscope drifts with time due to integration and needs a high pass filter. The accelerometer readings from (14) (has to be converted to degrees) and the gyroscope rate readings in deg/sec are passed through a complementary filter to mitigate noise and drift (see figure 2). (15) shows the complementary filter for pitch and (16) for roll (see Appendix B).

Calibration: Since different users will have different neutral/natural head orientation, it makes sense to calibrate the readings for each user. After the system setup, with the user's head upright and facing forward the micro-controller receives on notification, N IMU readings and takes the sample mean of these values which we call offset readings (see Appendix C) as shown in (18). With (19) and (20) we get the estimated pitch and roll angles, for which the neutral/natural head orientation gives values approximately equal to zero.

[4]BLEDevice Library: https://github.com/nkolban/ESP32_BLE_Arduino/blob/master/src/BLEDevice.h

[5]eSense-BLE-Specification: http://www.esense.io/share/eSense-BLE-Specification.pdf

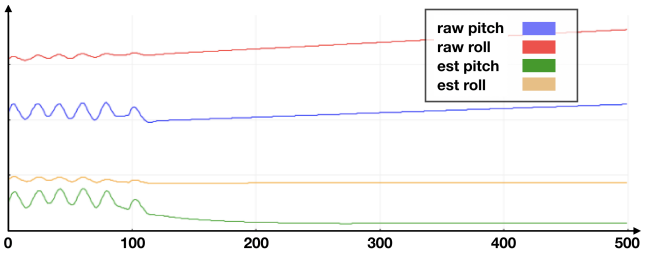


Figure 2: Raw pitch and roll angles calculated from the raw eSense IMU readings, showing drift compared to estimated pitch and roll angles after the complementary filter, with noise and drift eliminated. During samples 0-100, the user moved her head in the pitch direction, afterwards the user's head was held still.

Mapping: From (8) we observe that the yaw angle is eliminated during calculation, [3] because accelerometer cannot detect yaw rotations, since the G vector does not change during yaw. Hence, we have mapped the user's head estimated roll angle to the robot arm yaw motion (base servo motor, for left and right motion) and the user's head estimated pitch angle to robot arm pitch motion (for up and down motion). However, when the user's head makes a pitch motion, there is a bit of roll motion and vice versa (see Figure 2), hence, constraint/condition statements are used to determine exactly which motion(pitch or roll) is dominate/intended by the user before mapping.

3 EXPERIMENT DESIGN AND OUTLOOK

Our experiments are currently ongoing, where novice users are wearing the eSense to control the robot to mark (with a pencil) targets (as can be seen in Figure 1) as accurately and quickly as possible. Both performance measures are then taken over multiple sessions to also assess the learning of the interface. Intermediate results show that users tend to reach the targets from 10 centimetres within 5 seconds at an accuracy of approximately 9 mm from the target centre. What remains to be investigated is (1) an analysis of the importance of accuracy versus speed, and (2) the range of applications that such performances would allow.

REFERENCES

[1] Fahim Kawsar, Chulhong Min, Akhil Mathur, and Allesandro Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (jul 2018), 83–89.

[2] Hyung Gi Min and Eun Tae Jeung. 2015. Complementary filter design for angle estimation using mems accelerometer and gyroscope. *Department of Control and Instrumentation, Changwon National University, Changwon, Korea* (2015), 641–773.

[3] Mark Pedley. 2013. Tilt sensing using a three-axis accelerometer. *Freescale semiconductor application note* 1 (2013), 2012–2013.

[4] Nina Rudigkeit and Marion Gebhard. 2019. AMiCUS - A Head Motion-Based Interface for Control of an Assistive Robot. *Sensors* 19, 12 (2019), 2836. https://doi.org/10.3390/s19122836

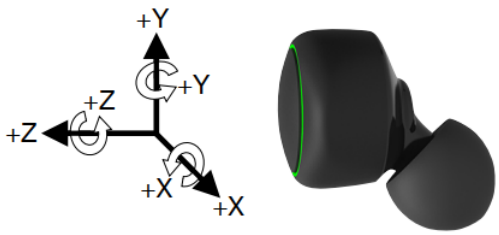


Figure 3: (Image from eSense User Documentation) The orientation of the IMU in eSense is shown above, the lower case of the axes are used in our calculations;x=+X,y =+Y,z=+Z. CCW direction on y axis represent Yaw direction, CCW on z axis represent Pitch direction and CCW direction on x axis represent Roll direction.

A EARTH REFERENCE FRAME TO BODY (ESENSE IMU) FRAME TRANSFORMATION

.

Similarly to [3], we use a transformation matrix R as shown in (4). We assume the accelerometer undergoes no linear acceleration and only experiences a gravitational field g, also the IMU y-axis is aligned to direction of g hence the unit vector (0,1,0) as shown in (5). The three-axis accelerometer reading is given by the vector G. We make use of the Euler angles; roll (about x-axis), pitch (about z-axis), and yaw (about y-axis) order for the transformation matrix;

$$R = R_x(-\phi).R_z(-\theta).R_y(-\varphi)$$

In (6), the angles are negative because we make a transformation from earth's reference frame to the IMU body frame.

$$\mathrm{G} = \begin{pmatrix} G_x \\ G_y \\ G_z \end{pmatrix} = R(g-a) \tag{4}$$

$$\mathrm{G} = \begin{pmatrix} G_x \\ G_y \\ G_z \end{pmatrix} = Rg = R\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{5}$$

$$\mathrm{R}\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = R_x(-\phi).R_z(-\theta).R_y(-\varphi).\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{6}$$

(7) shows the elementary rotations from earth's (inertial) reference frame to the body (IMU) frame.

$$\begin{aligned} R_x(-\phi) &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(-\phi) & -\sin(-\phi) \\ 0 & \sin(-\phi) & \cos(-\phi) \end{pmatrix} \\ R_z(-\theta) &= \begin{pmatrix} \cos(-\theta) & -\sin(-\theta) & 0 \\ \sin(-\theta) & \cos(-\theta) & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ R_y(-\varphi) &= \begin{pmatrix} \cos(-\varphi) & 0 & \sin(-\varphi) \\ 0 & 1 & 0 \\ -\sin(-\varphi) & 0 & \cos(-\varphi) \end{pmatrix} \end{aligned} \tag{7}$$

After applying (7) in equation (6), we arrive at (8) and (9) below,

$$\mathrm{R}\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -\sin(-\theta) \\ \cos(-\phi)\cos(-\theta) \\ \sin(-\phi)\cos(-\theta) \end{pmatrix} \tag{8}$$

which relates the transformed acceleration vector to accelerometer acceleration measurement in normalized form.

$$\hat{G} = Accelerometer\ measurements\ in\ normalized form$$

$$\hat{G} = \frac{G}{\sqrt{G_x^2+G_y^2+G_z^2}} = \begin{pmatrix} \hat{G_x} \\ \hat{G_y} \\ \hat{G_z} \end{pmatrix} = \begin{pmatrix} -\sin(-\theta) \\ \cos(-\phi)\cos(-\theta) \\ \sin(-\phi)\cos(-\theta) \end{pmatrix} \tag{9}$$

To get the pitch angle, we make use of the method in (10) that allows us to use arctan instead of arcsin , to avoid multiple angles for same value of acceleration measurement.

$$\begin{aligned} \frac{\hat{G_x}}{\sqrt{\hat{G_y}^2+\hat{G_z}^2}} &= \frac{-\sin(-\theta)}{\sqrt{\cos^2(-\theta)\left(\left(\cos^2(-\phi)+\sin^2(-\phi)\right)\right.}} \\ &= -\tan(-\theta) \end{aligned} \tag{10}$$

The pitch angle can then be calculated using (11).

$$\theta = -\tan^{-1}\left(\frac{-\hat{G_x}}{\sqrt{\hat{G_y}^2+\hat{G_z}^2}}\right) \tag{11}$$

Similarly, for roll, we get the relationship in (12) and can calculate roll angle using (13).

$$\begin{aligned} \frac{\hat{G_z}}{\hat{G_y}} &= \frac{\sin(-\phi)\cos(-\theta)}{\cos(-\phi)\cos(-\theta)} \\ &= \tan(-\phi) \end{aligned} \tag{12}$$

$$\phi = -\tan^{-1}\left(\frac{\hat{G_z}}{\hat{G_y}}\right) \tag{13}$$

To avoid issues that result with arctan, such as division by zero and inability to distinguish quadrants, we make use of atan2 in actual programming as shown in (14). Also, since the norm of G cancels out in (11) and (13), we can use the values of G non-normalized. The angles here are in radians.

$$\theta = -\mathrm{atan2}\left(-G_x, \sqrt{G_y^2+G_z^2}\right), \phi = -atan2\left(G_z, G_y\right) \tag{14}$$

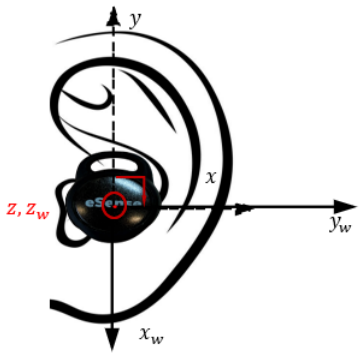


Figure 4: (Image from eSense User Documentation) eSense orientation when worn is approximately 90 degrees CW rotated about the z-axis from original orientation (see figure 3) used in calculation. the new axis orientation are in lower case *w*.

B COMPLIMENTARY FILTER

.

$$\theta_f(k) = \left(\theta_f(k-1) + gyro_z * \Delta t\right) * \alpha + \theta * (1-\alpha) \quad (15)$$

$$\phi_f(k) = \left(\phi_f(k-1) + gyro_x * \Delta t\right) * \alpha + \phi * (1-\alpha) \quad (16)$$

$$\alpha = \frac{T_s}{T_s + \Delta t}, T_s = \frac{1}{2\pi f_c} \quad (17)$$

$\theta_f(k)$, $\phi_f(k)$ *are current filtered pitch and roll angles*

$\theta_f(k-1)$, $\phi_f(k-1)$, *filtered pitch and roll angles* 1 Δt *in the past*

α , $(1-\alpha)$ *are the filter coefficients(weights) and sum up to* 1

$$T_s = \ filter \ \ time \ \ constant$$

$\Delta t = \ sampling \ \ time$, *determined by the IMU sampling rate*

$$f_c = \ filter \ \ cutoff \ \ frequency$$

The filter cutoff frequency is usually chosen such that the weight/coefficient of the gyroscope reading are favoured (greater) more than the weight of the accelerometer readings (since gyroscope readings are more accurate than accelerometer readings in short time intervals e.g. sampling time), and then tuned for best performance. The methods in [2] can be used to determine the filter coefficients.

C CALIBRATION EQUATIONS

.

$$Pitch\_offset = \frac{\sum_{k=1}^{N} \theta_f(k)}{N}$$
$$Roll\_offset = \frac{\sum_{k=1}^{N} \phi_f(k)}{N} \quad (18)$$

$$\text{Estimated\_Pitch\_Angle} = \theta_f(k) - Pitch\_offset \quad (19)$$

$$\text{Estimated\_Roll\_Angle} = \phi_f(k) - Roll\_offset \quad (20)$$

A data sharing platform for earables research

Jovan Powar
jsp50@cam.ac.uk
University of Cambridge

Alastair R. Beresford
arb33@cam.ac.uk
University of Cambridge

ABSTRACT

Ear-worn wearable devices, or *earables*, are a rapidly emerging sensor platform, with unique opportunities to collect a wide variety of sensor data, and build systems with novel human-computer interaction components. At this point in the development of the field, with projects such as eSense putting hardware in researchers' hands but being limited in reach, the sharing of datasets collected by researchers with the wider community would bring a number of benefits. A central data sharing platform would enable wider participation in earables research and improve the quality of projects, as well as being a vehicle for better data quality and data protection practices. We discuss the considerations behind building such a platform, and propose an architecture that would achieve better privacy-utility trade-offs than many existing data sharing efforts.

KEYWORDS

datasets, earables, wearable computing, pervasive computing, data sharing, privacy-preserving data sharing

ACM Reference Format:
Jovan Powar and Alastair R. Beresford. 2019. A data sharing platform for earables research. In *EarComp '19, September 07–09, 2019, London, UK*. ACM, New York, NY, USA, 6 pages. https://doi.org/xx

1 INTRODUCTION

The recent emergence of ear-worn wearable devices, or earables, presents novel opportunities for innovation and research in personalised computing, both through new modalities of human-computer interaction and as a sensor platform. Thanks to recent advances in earable computing power and commercial availability, we find ourselves at the beginning of a new branch of wearables research. In this work we propose a tool for this new community to share data and results from studies conducted with earables, with the aim to foster collaboration and lower the barrier to entry for new research, and establish and facilitate a set of standards for data quality and protection. A common system for data sharing would confer many benefits to the community at large: faster access to hard-to-collect data, diversity of samples and studies, and the ability to reproduce work to name but a few.

We aim to contribute a novel design for data sharing, one which enables sharing of rich and potentially sensitive datasets, integrating data gathered from across heterogeneous set of contributors and methodologies. We see the potential to introduce an approach to data sharing that strikes a better balance between utility and data protection than has been achieved in previous data sharing platforms.

In this paper we describe the types of data that can be collected with an earable platform, and discuss the potential breadth and implicit data protection burdens of data collection studies. Through a discussion of the fit of existing data sharing approaches to the earable case, and a wider range of considerations for a data sharing exercise, we propose a sharing platform design that accommodates heterogeneous studies with an adaptive, rule-based approach to data protection.

As the earables research community is in its infancy, there are many tradeoffs to be made between the incentives of contributors and third parties, as well as assumptions to be validated around the nature of future research. Therefore, we also present an appraisal of the potential benefits and drawbacks inherent in the data sharing problem, and invite members of the community to comment on these problems, many of which we leave open, in order to better shape the design of the system.

2 RESEARCH WITH EARABLES DATA

Earables are positioned on the head and next to the human sensory organs for sound, sight, speech, taste and balance—this makes them a platform suitable for collecting a wide variety of data. Motion or vibration sensors can collect data on activity, gait, speech, breathing patterns, or even facial expression. Sensors in contact with the skin can contribute continuous data on the internal state of the human body: optical sensors can measure heart rate and blood oxygenation, while electrodes measuring galvanic skin response can provide an indicator for stress. For now, we focus on the eSense project [3, 4], whose wireless earphone hardware

EarComp '19, September 07–09, 2019, London, UK

ACM ISBN ?...$00.00
https://doi.org/xx

incorporates a microphone and an inertial measurement unit (IMU).

The potential breadth of datasets collected by earables is exceptionally large, and will only continue to grow as new or improved sensor hardware is introduced. This phenomenon presents opportunities and challenges alike for a research community. For a data source of such breadth, we must anticipate that the applications of this data will be similarly broad, and cannot be predicted. As such, it is prudent to make available to future researchers as much and as varied a corpus of datasets as possible. This is especially important at the initial stages of earables research, where hardware is scarce. However, this also presents problems of data quality—when data is contributed by a highly heterogeneous set of initial collection studies, we must establish a baseline of documentation and quality standards.

Collection study design

We cannot anticipate all datatypes collected in studies with earables, and since the goal of the data sharing platform is to foster collaboration on novel research, the architecture of the platform must not—as far as is reasonable—impose restrictions on what datasets can be contributed. Instead, we propose that the system be designed to receive broad classes of data (discussed below in the context of privacy risks), upon which sharing and data protection policies can be designed. If a contributing researcher uploads a dataset with a datatype that is not included in the existing policies, it can be flagged for review—at which point the moderators of the system can perform a data protection analysis, leading to a new policy.

Studies will also differ in the modalities of collection. In terms of data collection 'episodes', being one continuous recording of earable sensor data from one subject, we should expect a wide variance both in temporal scope and environmental scope. Temporally, we expect datasets comprised both of short- and long-lived episodes of data collection, and datasets comprised of many one-off episodes or repeated episodes from a subject.

We expect datasets to span a range of environments, which can be most helpfully parametrised by the level to which they are controlled. This ranges from highly controlled—a lab environment where the subject performs specified tasks—to minimally controlled, or 'in the wild'—episodes that take place in unspecified public spaces, with no preordained activity being undertaken.

Parametrising this space of episode types will be useful both for ensuring utility—the third-party researcher using the platform will be able to easily find and compare similar datasets—and aid in data protection, as the sensitivity of a dataset can be highly dependent on temporal and environmental scope.

Data protection

With the richness of earables data comes a range of implicit data protection burdens. A central challenge for any data sharing system will be to support and manage the *greatest* burdens associated with current or future datatypes, while *minimising the friction to third-party researchers* who wish to make use of the datasets. Below we outline the data protection burdens implicit to a range of earable-collected datatype.

We propose that a data sharing platform for earable data should be capable of receiving datasets containing any or all of these datatypes, and tailor its data protection processes adaptively to which datatypes are included—either at contribution-time or at query-time. This way, greater procedural friction associated with one datatype, such as audio, does not need to be applied to a third-party who wishes only to access short-term IMU logs.

IMU and mobility. Privacy risks from mobility data and IMU (Inertial Measurement Unit) traces are usually minimal in the average case for research, where only short-lived traces are collected. Even in these cases, these traces may confer personal but non-identifying information about the participant, such as the presence of Parkinson's tremors. Similarly, long-term traces leak information about activity such as commute timings, leisure activities, or working schedules. Where study design cannot obviate such latent indicators, care must be taken to constrain analysis such that this information is not misused.

Some reidentification risk may come from the uniqueness of certain mobility characteristics such as gait, which has been used to fingerprint individuals [6]. It is unclear how much entropy can be derived from ear-collected gait analysis, but it is unlikely that it could be used to 'blindly' identify a subject—that is, if one does not already have a gait fingerprint of the subject, and does not already know that they contributed to the dataset. Under a conservative evaluation of the reidentification power of IMU data, it will nonetheless be necessary for contributors to assume that if a study participant contributes IMU data to their dataset, third parties will be able to reidentify that participant in other datasets available through the platform. This must be considered in any consent agreements made between contributors and their subjects.

Audio. As earables are usually marketed to users as wireless headphones, they invariably include a microphone. Audio recordings are a potentially highly sensitive type of data to collect on subjects, especially if the study takes place 'in the wild'. This problem is compounded by the fact that we cannot know for sure whether a recording contains sensitive information (e.g. whether the recording contains a personal conversation including the subject or a conversation between

others in the same space) without listening through it completely.

Therefore, a data sharing platform must allow contributors to tag their datasets with information that describes the risk of this sensitivity. If the dataset is tagged as containing only short audio snippets from a controlled environment, minimal data protection mechanisms would need to be employed; if the recordings were made on the street, or at the subject's home, the system must consider each of those cases as progressively more sensitive, and apply greater protections. These might come in the form of stronger licensing agreements, stronger consent requirements to contribute data at all, or automated transformation of audio into representations with lower fidelity.

Future sensors (e.g. electrodes). While we can speculate about the usage and collection modalities of novel sensor datatypes, such as galvanic skin response, we know that the uses of those datatypes will evolve as they become available to researchers. Therefore, it is important not to prescribe data protection policies for these datatypes but to continuously evaluate the tradeoffs between their sensitivity and their utility.

While the system's policies must be incrementally formed as more datatypes are added to the corpus, it is important that this early lack of strategy be properly presented to subjects at the point of consent. Consent documentation must clearly explain the open-ended nature of the usage of the subjects' data, and the subjects' right to have the data minimised or better protected as soon as technical means become available should be communicated.

Metadata. Any earables research study will contain study-level and subject-level metadata. Both to ensure utility to third-parties and for data protection, it is important to establish a baseline of data quality for these sorts of information. The question of how strictly to draw the specification of this metadata should be agreed with the community.

In the case of study-level metadata, a majority of datasets contributed to a public repository will not share collection methodologies or pre-processing approaches, even in the initial case where we focus solely on the eSense platform. Therefore it is necessary to provide detailed metadata on the activity captured by each dataset. This must include, but not be limited to the time-frame of collection episodes, their frequency, and the degree to which the environment was controlled. As noted above, each of these will also contribute information on the sensitivity of the dataset.

At the subject-level, contributors will want to include data such as age, gender, or level of physical activity, or specific tagging of the sensor data collected, such as by activity or location.

Accessory data. In addition to earable-collected data, contributors can be expected to collect concurrent data streams from other sensors, such as smartphones or fitness trackers. Any platform must be able to host this data alongside earable datasets. This linkage will likely often increase the sensitivity of the dataset and hence the protection burden, and so we propose that a rules-based approach be taken to granting access to these datasets, using an evaluation of the extra privacy risk created by the linkage.

3 EXISTING APPROACHES TO SHARING DATASETS FOR RESEARCH

There are a number of other data collection projects which have made their data available to third-party researchers, as well as systems designed to handle that sharing. These studies range in scope from collection studies where the researchers have collected a dataset and roll their own sharing platform, to collection and sharing tools created for third parties to integrate into their own studies, to platforms that simply serve as repositories for datasets (employing varying levels of mediation).

Two notable collection studies are the Device Analyzer [7] and Haystack [1] projects. Each of these studies publicly released an Android app to capture various data about an individual's smartphone activity. These subjects are members of the public who are incentivised by reports about their smartphone activity provided by the app, as well as a desire to contribute to scientific research.

These studies took a different approach to sharing their data. The Haystack project published a dataset containing a subset of the data collected—anonymised TLS handshakes for 1378 devices over the course of two years. This anonymised data was published under a Creative Commons Attribution 4.0 International license and used a public dataset sharing website Zenodo.

Device Analyzer, on the other hand, makes available their entire dataset, after post-processing and pseudonymisation of certain fields [8]. Since the data may contain sensitive information even when pseudonymised, the sharing mechanism chosen employed significantly more friction. In particular, third-parties were required to submit a research proposal and arrange a license agreement between the University of Cambridge (who hosted the platform) and the third-party's institution. This process sometimes took weeks or months.

Both of these approaches are instructive for the design of an earables sharing platform, but neither can be followed closely. Haystack's approach allows a low barrier to entry, but only for a very limited subset of their data, while Device Analyzer provides a rich dataset but behind a significant barrier to entry (by design). While these projects illustrate very different points on the privacy vs. utility spectrum, they

do not match our case exactly as they deal with only one collection study. A platform for earables, which could host data from zero to high sensitivity, would be of little use to the community if it were to adopt a one-size-fits-all approach: the Device Analyzer model is prohibitively constrictive, while publishing on a general-purpose platform provides inconsistent data protection practices, and precludes tailored solutions for earables research such as automated pre-processing and tagging. Instead, we advocate a system that can adapt to the sensitivity of datasets with progressively stronger or more suitable protections.

The AWARE framework [2] provides a platform for researchers to have study participants contribute data from their smartphone, such as ambient noise or location. Researchers can write their own plugins to collect specific data for their purposes. The data is uploaded to the servers (hosted by the project or by the researchers themselves), which host a web dashboard for researchers to access the data. Collection tools such as AWARE are able to provide a single interface to sensors which can be used by different researchers, delivering a uniformity of data format and quality that is useful for inspecting data from multiple studies.

There are a number of systems proposed for hosting data securely and privately. One notable recent work is ScrambleDB [5], a database designed to store multiple datasets, as is our goal. Its 'pseudonymisation-as-a-service' allows a dataset to be decoupled into constituent datasets, providing non-linkability guarantees for the decoupled outputs. In addition to enforcing access control, this principle allows a more granular, targeted enforcement of data protection policies. Approaches such as these are central to our proposed system, as is detailed in Section 5.

4 PARAMETERS ON DATA SHARING

There are many parameters for a data sharing platform that must be set at the time of design. Here we present a range of parameters relevant to the earables case; we present our reasoning around each, but for the most part we leave the parameters open to input from the wider earables community.

Third party qualification

Who? As noted in the comparison between Device Analyzer and Haystack in Section 3, there are a range of options for who should be allowed to browse and download datasets from the platform. The four options we consider, in increasing order of restrictiveness are: (1) anyone, via a public website, (2) anyone who makes an account on the platform and agrees to Terms of Service, (3) only with researchers intending to pursue work for public academic publication (including industrial researchers); and (4) only with academic researchers at known institutions.

Each increased level of restriction serves two purposes: first, 'locking in' researchers to the platform aids community building, as it increases the likelihood that they will comply with terms of service or license agreements; second, increasing the likelihood of compliance serves as a data protection mechanism, which can alleviate the need to perform more restrictive technical transformations—put simply, the more likely the third party is to behave well, the more of the raw data you can give them.

We believe that a third party should need an account to use functionality beyond browsing metadata. A generic license agreement, covering citation rights and guidelines on ethical usage of the data, should be agreed to before a third-party is allowed to view any datasets.

From a legal perspective, it may be the case that for certain datatypes sharing will be covered by the GDPR. Here we have two options: prohibit upload of those datatypes, or limit third parties to researchers in GDPR-compliant countries. Until we clarify this legal situation, it would be prudent to take the last option.

With a community resource, there is always a danger that someone may take without giving; the more often third parties withdraw data without depositing new data, the more likely the system is to suffer abuse. This could be combatted by only allowing third-parties to check out datasets if they have already shared their own. However, this policy is also vulnerable to abuse, as it may encourage users to contribute fake or incomplete data. We leave open the question of how to incentivise good community behaviour.

Datatypes

As noted in Section 2, the datatypes collected as part of earables research are diverse and cannot be anticipated. Therefore, it is important that any sharing system aiming to facilitate future research is able to host arbitrary datatypes. Currently, we will focus on the eSense platform, which has an IMU onboard, as well as a microphone; even in this case, it would be overly prohibitive to allow only IMU traces and audio recordings to be uploaded. Studies will likely generate much more varied data, such as information about the collection environment, participants, location, etc.

Expanding the scope of data handled by the platform significantly complicates the problem of data protection; we discuss our approach to this problem in Section 5.

An open question is how to ensure data quality. By specifying data representations, or accepting only aggregate data, we impose a degree of homogeneity on contributors' data, which may limit the design of contributors' studies and may turn them off sharing data altogether. Conversely, if we were to also provide libraries to be used in data collection (similar to the AWARE framework), we might lower the barrier to entry for new research.

Tooling complexity

If successful, the platform would be well placed to do some heavy lifting on the behalf of researchers. This might be as simple as producing aggregate data for export, or as complex as hosting and running analysis code. Both of these tooling options provide utility while also improving data protection; contributors can specify that only aggregate data may be used by third parties rather than allowing downloads of raw data, or the third party could never be given the data at all, instead having their code 'come to the data'. However, this might be unnecessary complexity—it remains an open question what degree of use such a system would garner.

The obverse approach would be to not host the data on the platform at all—it may well be the case that even preparing the data for upload is excessive work for a contributor, and they would prefer instead a catalogue, where third parties can find their details and request the data in whatever way the contributor sees fit. This would, of course, negate much of the data protection benefits we have discussed, along with the secondary benefits to the community. We discuss incentives further in Section 6.

5 A PROPOSED ARCHITECTURE

We propose a system in which contributing researchers upload their data to a shared platform (either centralised or instances of a common software stack) as a collection of linked datasets, one collection per study. Datasets are split into two classes: **core** sensor data and metadata and **accessory** non-sensor data or additional metadata.

Core datasets

The initial classes of core sensor dataset will be IMU data and audio recordings. For each, we will require each collection episode is tagged with a pseudonym for the subject, a timestamp, and appropriate metadata such as sampling frequency. A contributor must also describe the details of the study methodology—the temporal and environmental scope of collection, as well as other study-level metadata as described in Section 2.

Accessory datasets

Accessory datasets are any further information collected as part of a study. Examples might include detailed subject information, location traces, and ground truth or concurrent data collected from other sensors.

Checkout

A third-party researcher wishing to check data out of the system will be presented with an interface to browse studies and their datasets, inspect metadata, and choose which datasets they wish to download using a 'checkout' model. At this point, any procedural or legal data protection mechanisms can be applied—agreeing to further license agreements, verifying institutional affiliation etc.

The third-party will also be given a metadata summary of the dataset they will be supplied—statistics such as sampling frequencies and included columns. This summary will give an overview of the output, including the level of availability of particular fields if the output has been composed from multiple studies. It also serves to highlight where transformations have been applied to the dataset due to data protection policies.

Protective transformations and limitations

Depending on the permissions granted to a third party, and the data that they have checked out (or are requesting to check out), transformations may be applied to core datasets to minimise data protection risks. Which protections are enforced and when is specified by global system policies—some globally applied and some discretionary on the part of the dataset's contributor. For each class of core or accessory dataset, there must be a set of such policies; in the event that a novel datatype is collected, an expert evaluation would be performed to create new policies to address any new challenges.

For example, audio data requires global policies for desensitisation: the platform should automatically perform different transformations of raw data into less sensitive representations, with stricter access control for greater fidelity.

Examples of discretionary policies include: (1) the dataset's contributor has specified that third parties with a different institutional affiliation to theirs must receive a lower IMU sampling rate (2) due to the consent form given to participants, you must have agreed to a license agreement with the contributor to receive column X in this table (3) you have previously checked out a location trace dataset from this study, and so cannot be provided with this audio trace dataset (4) you must submit a research proposal to the dataset's contributor to be granted access to this study

6 COMMUNITY INCENTIVES

As the sharing platform we propose is intended to facilitate the growth of a research community around earables, we must consider the balance of incentives of our target users, as well as validate the assumptions we have made about their intents. We assume, for example, that the majority of contributors would be happy to share their data were it a simple enough task, and that there is significant utility to be gained from increasing discoverability and allowing composition of different datasets. We also assume that an earables-collected dataset will have utility to future studies. We expect that the 'first wave' of eSense research will help us validate these assumptions.

We here present a rundown of the incentives we anticipate to influence uptake and usage of an earables data sharing platform. We discuss each incentive and disincentive to the best of our ability, but we require input from the research community to validate them and understand the true balance between them.

Motivations to contribute data

Community kudos. Contributors who have produced a novel or high quality dataset would like recognition from peers, and a higher chance of being cited.

Allow subjects access to their own data. The platform could alleviate the need for contributors to build an access portal where their study's subjects can view their own data.

Easy compliance. Contributors could use carefully designed systems for managing data protection, with subjects presented a pre-prepared consent process. This would reduce the work contributors need to do, and assure compliance with the GDPR.

Reproducibility. Making data available means that others can verify any published analysis. Some funding bodies require study data is made available for this reason.

Motivations not to contribute data

Perceived additional work. If the contributor wishes to collect data in a specific way, it may be perceived as too much effort to make their dataset compliant with whatever standards the platform expects. The prospect of having to spend time tagging and uploading may also seem overly onerous.

Existing non-compliant practices. A prospective contributor may already have collected a significant portion of their dataset without having obtained sufficient consent from participants for further sharing. In this case, it is unlikely the contributor will discard that data. Similarly, if the platform's data protection rules and consent requirements are stricter than those of the contributor's institution, the extra work may be perceived as overly restrictive.

Institutional wariness. In the case of a sharing architecture that hosts data at an institution other than the contributor's, they may have reservations—either because they do not want to share with the hosting institution, or do not trust that institution's processes to manage sharing with others. This wariness could be enough for some researchers that they would prefer to share directly with a third party.

Reverse kudos. If the contributor's study ended inconclusively or if the data was of poor quality, the contributor may choose not to publicise their data for fear of judgment (even though that dataset might still be useful for other purposes).

Motivations to check data out

Data availability. If successful, the platform would host a wide diversity of samples, and a larger volume of data than even a well-resourced researcher could easily collect.

Easier than collecting fresh data. If a third-party wishes to quickly test an assumption, or has limited resources, it would often be easier to check out a dataset from the platform, rather than collecting it independently.

Unavailable accessory data. Similarly, a third-party could find datasets with accessory data that they would not have been able to collect themselves, such as high-fidelity ground truth IMU, heart rate, or room temperature.

7 CONCLUSION

In this paper we have proposed a data sharing platform for earables research, and discussed the considerations involved in ensuring it provides utility to the emerging earables research community. This approach could simplify many aspects of running studies—such as providing a streamlined consent process and handling legal compliance. We believe our proposed architecture allows for better balancing of collaborative utility and data protection than previous efforts.

We have left a number of questions open for discussion, most notably: (1) Should the platform include standardised data collection tooling? (2) How strictly should data format and quality standards be specified? (3) Do the perceived benefits of a complex sharing system outweigh the drawbacks? (4) Does the community see utility in this approach? (5) How do we encourage users to be good community actors?

REFERENCES

[1] ICSI UC Berkeley. last accessed 2019-08-15. ICSI Haystack Project. *https://haystack.mobi/.*

[2] AWARE framework. last accessed 2019-08-15. AWARE - Open-source Context Instrumentation Framework For Everyone. *http://www.awareframework.com/.*

[3] F. Kawsar, C. Min, A. Mathur, and A. Montanari. 2018. Earables for Personal-Scale Behavior Analytics. *IEEE Pervasive Computing* 17, 3 (Jul 2018), 83–89. https://doi.org/10.1109/MPRV.2018.03367740

[4] Nokia Bell Labs. last accessed 2019-08-15. eSense Earable Computing Research. *http://www.esense.io/.*

[5] Anja Lehmann. 2019. ScrambleDB: Oblivious (Chameleon) Pseudonymization-as-a-Service. *Proceedings on Privacy Enhancing Technologies* 2019, 3 (2019), 289–309.

[6] Hoang Minh Thang, Vo Quang Viet, Nguyen Dinh Thuc, and Deokjai Choi. 2012. Gait identification using accelerometer on mobile phone. In *2012 International Conference on Control, Automation and Information Sciences (ICCAIS)*. IEEE, 344–348.

[7] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. 2014. Device Analyzer: Large-scale mobile data collection. *ACM SIGMETRICS Performance Evaluation Review* 41, 4 (2014), 53–56.

[8] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. last accessed 2019-08-15. Device Analyzer for Android. *https://deviceanalyzer.cl.cam.ac.uk/.*