Recognizing Human Activities from Smartphone Sensor Signals

Arindam Ghosh
DISI
University of Trento
Trento, Italy
aghosh@disi.unitn.it

Giuseppe Riccardi
DISI
University of Trento
Trento, Italy
riccardi@disi.unitn.it

ABSTRACT

In context-aware computing, Human Activity Recognition (HAR) aims to understand the current activity of users from their connected sensors. Smartphones with their various sensors are opening a new frontier in building human-centered applications for understanding users' personal and world contexts. While in-lab and controlled activity recognition systems have yielded very good results, they do not perform well under in-the-wild scenarios. The objective of this paper is to 1) Investigate how audio signal can complement and improve other on-board sensors (accelerometer and gyroscope) for activity recognition; 2) Design and evaluate the fusion of such multiple signal streams to optimize performance and sampling rate. We show that fusion of these signal streams, including audio, achieves high performance even at very low sampling rates; 3) Evaluate the performance of the multi-stream human activity recognition under different real end-user activity conditions.

Categories and Subject Descriptors

I.5.4 [**Pattern Recognition**]: Applications—*Signal processing*; H.1.2 [**Information Systems**]: User/Machine Systems—*Human factors*

Keywords

Human Activity Recognition; Signal Fusion; Smartphone Sensing

1. INTRODUCTION

In context-aware computing, Human Activity Recognition (HAR) aims to understand the current activity of users from their connected sensors. Portable sensors platforms are either wearable or smartphone-based. In the first case, wearable computers are designed to deliver target signals such as Heart Rate, Skin Temperature, Galvanic Skin Response. In the second case smartphone platforms are usually designed to support sensors such as gyroscope, accelerometer, microphone. These sensors can be used for understanding the user's activity which can provide us with contextual information about the user's present state. Knowing whether a user is sitting or walking or commuting can significantly affect human-computer interaction. Activity Recognition has been used in various fields from monitoring the elderly [13] to providing contextual recommendations [2].

Activity recognition systems can be classified as vision based, sensor based, or a fusion of the two. While vision based systems use external cameras [16] to monitor the users and hence are limited in their application scenario, sensor based systems are usually wearable and can monitor users in more unconstrained environments. Till recent years in most sensor-based approaches, users had to attach multiple dedicated motion sensors [11] to various parts of the body such as legs, arms, and waist. While such systems have been able to achieve high recognition performances, they require elaborate set up and can be uncomfortable to wear and hence are not very suitable for long-term monitoring.

The recent proliferation of smartphones with their plethora of sensors have opened up a new frontier in context aware human computer interaction. Most modern smartphones have embedded sensors such as microphone, camera, accelerometer and gyroscope. Accelerometer and gyroscope sensors have been successfully used to detect human activity [1], understand human mobility patterns [4], and monitor Activities of Daily Living [17]. Scientists have also exploited the microphone on smartphones for daily activity recognition. The SoundSense [10] project used the microphone on a smartphone for detecting and modeling sound events in everyday life. Bieber et al [3] combined the accelerometer and microphone sensors for detecting everyday activities. Schuller et al in [15] used the microphone of a smartphone for acoustic geo-sensing to automatically determine a cyclists route.

However, smartphone sensors are not robust, and performance of a single-sensor based classification systems leads to sub-optimal and non-robust performances. The quality and response of on-board accelerometer and gyroscope sensors vary across manufacturers and devices. Certain smartphones do not have dedicated gyroscope hardware, and implement it in software propagating errors from accelerometer into gyroscope readings. The accelerometer performance for activity recognition task degrades rapidly when the user is playing a game on the smartphone or using an application. Similarly, as reported in [10] audio data cannot be the sole source of information when the phone is in a backpack or the

MM'14, November 3–7, 2014, Orlando, Florida, USA.

http://dx.doi.org/10.1145/2647868.2655034.

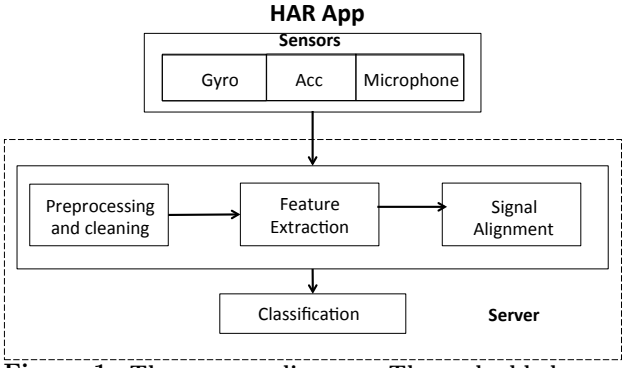

Figure 1: The process diagram. The embedded smartphone sensors record data and stream them to the server where pre-processing, feature-extraction and alignment, and classification steps are performed.

user is on a call. The solution is to use multiple weak signals and combine them to improve the recognition of a user's activity state. Combining multiple sensors can also lead to opportunistic sensing, thus improving the energy consumption of the phone by smart decisions on turning on/off sensors at appropriate times.

Proper evaluation of an activity recognition system is another challenge. Although activity recognition using smartphone data is a popular research field, very few publicly available corpora have raw data available. Even then, most of the corpora were created under controlled environments with static phone placements, or scripted activities where the user does not otherwise use the smartphone during the data collection. Therefore we carried out our own data collection on the Android and iOS platforms in a naturalistic settings. We also collected a smaller "stress-test" corpus where the data was collected while the participants were actively using the phone [1].

2. DATA COLLECTION

Two sets of data collection were performed to the hypotheses. Each experiment involved 15 (9 male and 6 female) participants. Participants varied in age between 25 and 40. The devices used were smartphones running android [2] (10 participants) and iOS[3] (5 participants) operating systems. Participants were located in various cities in Italy, Spain, and India, thus providing our data a wide variability. While a separate application was developed for each platform, both applications had the same functionality and recorded data from the same set of sensors. The applications sampled the *tri-axial accelerometer sensor, gyroscope sensor, location sensor, and microphone* during the data collection phase. The tri-axial accelerometer and gyroscope data was recorded at the rate of 40 Hz, and audio was recorded with a 8 KHz sampling rate. Location data was collected at the rate of 1 sample per 5 minutes, and was used only for validation and was not a part of the activity recognition process. Also, the participants often turned off location service on their phone to decrease power consumption.

The first data collection was a controlled scenario where the participant did not interact with the phone during the length of the experiment. Data was collected for six activities. They are: *Walking, Standing, Sitting, Driving, Travelling by bus, and Travelling by train.* The data collection protocol involved the participant launching the application before starting an activity, marking the activity start point on the application, and going on with the activity, finally marking the end when the activity was over. Participants were free to carry the phone as they wanted, but had to annotate the phone placement (pocket, purse, in hand, etc) at the start of the activity using a multiple choice drop-down in the application. The participants were asked to upload the data to our servers at the end of the day. From our data we observe that in majority of the cases, the phone was carried on the body (front or back trouser/jacket pocket), with 3 instances of the phone being placed in the purse. We ignored two instances where the participant was sitting and the phone was kept on the table.

The participants were free to delete sessions in case there was any undesired characteristic to the data. This could include personal data (audio,location) which the participants did not wish to share.

We collected approximately 31.6 hours of data (See Table 1), with each individual activity session ranging from 5 minutes (mostly walking) to 1 hour (commuting).

Walking	Standing	Sitting	Driving	By Bus	Train
4.12	8.31	8.23	3.13	2.19	5.12

Table 1: Activity Distribution in hours for normal activities. The participants placed the phones in pre-defined positions and did not interact with the phone during the experiment. Reported numbers are in hours.

Walking	Standing	Sitting	By car	By Bus	Train
0.32	1.21	3.23	2.19	2.47	1.3

Table 2: Activity Distribution for noisy activities. The participants were actively using their phones(playing games,texting or typing emails) during this experiment.Reported numbers are in hours.

For the second data collection, our goal was to collect data while the participants were actively using the phone. Most activity recognition experiments have low performance in real scenarios because they ignore the fact that people interact with their smartphones. During this data collection, the participants were asked to play games or type email or text messages for the duration of the data collection. We replaced the driving scenario with *Travelling by car* during this experiment. Approximately 10 hours of data (See Table 2) was collected for this scenario.

3. EXPERIMENTS AND RESULTS

3.1 Feature Extraction

To remove unwanted noise from the beginning and end of each activity session, we remove the first and last N seconds of the data. N was taken at T/10 seconds with a max value of N=30 seconds where T was the duration of the activity session. We only consider sessions which lasted at least 5 minutes for our experiments.

We divide each(Acc, and Gyro) signal channel into a 3 second sliding window with 50% overlap which has been shown to be effective [1, 14] window size for best performance in activity recognition using smartphones.

[1]The data is available for research purpose only. Please contact http://sisl.disi.unitn.it/

[2]version 4.0 and higher

[3]iOS 6 and higher

3.1.1 Accelerometer and Gyroscope

Accelerometer and Gyroscope each have 3 axes *x,y,z*. We first compute the acceleration magnitude, given by:

$$A_{norm} = \sqrt{{A_x}^2 + {A_y}^2 + {A_z}^2}$$

Now for each of the x, y, z, axes and norm of the accelerometer and gyroscope data we extract standard features for each 3-second window. We calculate the mean, standard deviation, min, max, number of peaks, number of zero crossings, inter-peak distances, etc for each of the accelerometer and gyroscope axes.

3.1.2 Audio

We use the same window size for processing and extracting features from the audio signal. While extracting features, we segment the audio stream into small uniform frames. Standard frame-sizes for audio processing lie between 25-46 milliseconds. In our case we use a 23 milliseconds half-overlapping subframes of audio as used by McKinney et al [12].

For the audio signal, we use Opensmile [7] to extract features. The following are the main features for each window:

1. Zero crossing rate - ZCR is defined as the number of time-domain zero-crossings within a frame.
2. RMS Energy - We use the Simple Moving Average of the mean, standard deviation, skewness, max, min, and range of the RMS energy of each window.
3. MFCCs which are very commonly used in Speech and Speaker recognition, have been recently used for recognition of environmental Sound [5]. We use the Simple Moving Average of the mean, standard deviation, skewness, max, min and range of 12 MFCCs for each window.

Durrent et al. [6] defines sensor fusion configuration as complementary if the sensors do not directly depend on each other. While older smartphones use a software gyroscope, modern smartphones (which were used for our experiments) have a dedicated gyroscope chip. So in our experiment we treat the sensor channels as complimentary and use the absolute time for each sensor event (recorded during our data collection) to align the data. We perform a feature-level fusion (early fusion) of the different streams by concatenating the time-aligned feature sets before the learning stage. For each classification experiment we perform feature-vector normalization before training. For each window, all feature-vectors form a $m \times n$ matrix where m is the *window size* $\times$ *sampling rate* for that window and n is the *length of each feature vector*. For each feature f_{ij} in the feature vector where i=1 ... n is the number of the feature ,and j is the jth row we normalize the feature using:

$$f_{ij} = \frac{f_{ij} - Min(f_{ij})}{Max(f_{ij}) - Min(f_{ij})}, i = 1 \ldots n; j = 1 \ldots m$$

3.2 Classification and Results

We used the WEKA machine learning toolkit [8] to test different classifiers using the above normalized features. First we used the data from the first experiment with the highest sampling rate to choose a classifier. We tested three different classifiers : Support Vector Machines, J48 decision trees, and random forests. Random forests provided us with the best F-measures (see Table 3), hence further classification was done with random forest for the different sets of experiments. For all classification experiments results were obtained by 15-fold (leave-one-subject-out) cross validation where each fold corresponds to the data for one subject. We used the full feature set with data sampled at 40 samples/second.

Signal	**Acc**	**Gyro**	**Audio**	Acc Gyro	Acc Gyro Audio
SVM	0.85	0.79	0.79	0.87	0.91
Decision Trees	0.85	0.82	0.83	0.85	0.90
Random Forest	**0.91**	**0.84**	**0.85**	**0.93**	**0.98**

Table 3: Average F-measure of 15-fold (one-fold-per-user) cross validation for the classification algorithms tested with the full feature set with data sampled at 40 samples/second and a 3-second sliding window

A major problem with using sensors is that polling them continuously can lead to power drain. One of the goals of our experiment was to determine how gracefully the recognition quality decreases when the sampling rate is decreased.

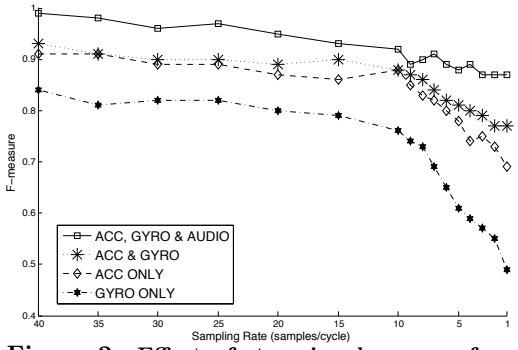

Figure 2: Effect of stepwise decrease of sampling rate (from 40 to 1 samples/sec) on F-measure. We see that adding audio leads to better accuracy at lower sampling of acc and gyro sensors.

We carried out LOSO cross-validation classification experiments using random forests. We experimented with different window sizes and a 3-second window was confirmed to be the best. Krause et. al in [9] showed that sampling rate of sensors has a direct effect on the battery life of a wearable device and decreasing sampling rate lowers power consumption. We performed stepwise downsampling of the accelerometer and gyroscope signals from 40 sample/sec to 1 sample/sec. From Figure 2 we see that accelerometer performance(F-measure) drops from 0.91 to 0.69 when the sampling rate is decreased from 40 to 1 sample/sec. Gyroscope performance (F-measure) drops more steeply from 0.84 to 0.49 in this range. A combination of Accelerometer and Gyroscope fairs comparatively better, degrading from 0.93 to 0.77. Adding audio signals from the microphone not only helps to provide better results at higher sampling rate (0.98 at 40 samples/sec), but also helps to balance the drop to only 0.89 at the lower end. However, we did not experiment with different sampling rates of the audio because of the limitation of the audio format for recording. AAC audio coding, which is the standard audio codec on both iOS and android does not support compression below 8 kHz. So all experiments with audio were carried out at this sampling rate.

From Table 4 we see that under controlled experimental conditions, accelerometer performance can be a good measure for understanding a participant's current motion pro-

Signal	Avg precision	Avg Recall	Avg F-measure
Acc	0.90±0.02	0.91±0.03	0.91±0.03
Gyro	0.83±0.08	0.84±0.07	0.84±0.08
Audio	0.85±0.06	0.86±0.07	0.85±0.07
Acc Gyro	0.93±0.06	0.92±0.06	0.93±0.06
Acc Gyro Audio	**0.98**±0.05	**0.97**±0.05	**0.98±0.05**

Table 4: Average Precision, Recall and F-measure using random forests. For this experiment the participants were requested to place the phones at pre-defined location and not use it during the experiment

Signal	Avg precision	Avg Recall	Avg F-measure
Acc	0.75±0.13	0.77±0.10	0.76±0.11
Gyro	0.70±0.17	0.74±0.16	0.72±0.16
Audio	0.85±0.08	0.85±0.09	0.85±0.08
Acc Gyro	0.79±0.13	0.80±0.12	0.80±0.13
Acc Gyro Audio	**0.86±0.11**	**0.88±0.11**	**0.87±0.12**

Table 5: Average Precision, Recall and F-measure of sensor channels using random forests. For this experiment the participants were actively using the phone during data collection

file. In this experiment the participants were expected not to interact with the phone for the duration of the experiment. While post-processing we ensured that we removed all instances where the screen of the phone was unlocked for durations longer than 10 seconds during a data collection session since it indicated that the participant was using the phone. However, since the participants were free to carry the phones as they wanted, this data collection is less controlled than other controlled data collection [1, 4] for activity recognition.

Table 5 shows the recognition results when the participants were actively using the device phone while performing an activity. Average precision for single sensor channels is lower than in the controlled experiments reported in Table 4 . The accelerometer and gyroscope individually perform lower (F-measures 0.76 and 0.72 respectively) than in the controlled scenario (F-measures 0.91 and 0.84 respectively). While combining the two sensor streams improves the recognition rates (F-mseasure 0.80), combining the motion sensor channels with audio achieves the best results (F-measure 0.87) under this scenario.

4. CONCLUSIONS

In this paper we have explored the use and combination of multiple weak smartphone sensors. We show that while smartphone sensor quality drops heavily during real-world usage, combining multiple signal streams can lead to better recognition results. We also exploit the less-used microphone sensor on a smartphone. We show that by leveraging audio features we can achieve high results. By combining audio features with other weak sensor features we can come up with a robust activity recognition scheme.

5. REFERENCES

[1] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz. Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Ambient Assisted Living and Home Care*, pages 216–223. Springer, 2012.

[2] V. Bellotti, B. Begole, E. H. Chi, N. Ducheneaut, J. Fang, E. Isaacs, T. King, M. W. Newman, K. Partridge, B. Price, et al. Activity-based serendipitous recommendations with the magitti mobile leisure guide. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1157–1166. ACM, 2008.

[3] G. Bieber, A. Luthardt, C. Peter, and B. Urban. The hearing trousers pocket: activity recognition by alternative sensors. In *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments*, page 44. ACM, 2011.

[4] Y. Chon, E. Talipov, H. Shin, and H. Cha. Mobility prediction-based smartphone energy optimization for everyday location monitoring. In *Proceedings of the 9th ACM conference on embedded networked sensor systems*, pages 82–95. ACM, 2011.

[5] S. Chu, S. Narayanan, and C.-C. Kuo. Environmental sound recognition with time–frequency audio features. *Audio, Speech, and Language Processing, IEEE Transactions on*, 17(6):1142–1158, 2009.

[6] H. F. Durrant-Whyte. Sensor models and multisensor integration. *The International Journal of Robotics Research*, 7(6):97–113, 1988.

[7] F. Eyben, M. Wöllmer, and B. Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the international conference on Multimedia*, pages 1459–1462. ACM, 2010.

[8] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[9] A. Krause, M. Ihmig, E. Rankin, D. Leong, S. Gupta, D. Siewiorek, A. Smailagic, M. Deisher, and U. Sengupta. Trading off prediction accuracy and power consumption for context-aware wearable computing. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 20–26. IEEE, 2005.

[10] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell. Soundsense: scalable sound sensing for people-centric applications on mobile phones. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 165–178. ACM, 2009.

[11] U. Maurer, A. Smailagic, D. P. Siewiorek, and M. Deisher. Activity recognition and monitoring using multiple sensors on different body positions. In *Wearable and Implantable Body Sensor Networks, 2006. BSN 2006. International Workshop on*, pages 4–pp. IEEE, 2006.

[12] M. F. McKinney and J. Breebaart. Features for audio and music classification. In *ISMIR*, volume 3, pages 151–158, 2003.

[13] G. Perolle, P. Fraisse, M. Mavros, and I. Etxeberria. Automatic fall detection and activity monitoring for elderly. *Proceedings of MEDETEL*, 2006.

[14] A. Reiss, G. Hendeby, and D. Stricker. A competitive approach for human activity recognition on smartphones. In *ESANN 2013*, pages 455–460. ESANN, 2013.

[15] B. Schuller, F. Pokorny, S. Ladstatter, M. Fellner, F. Graf, and L. Paletta. Acoustic geo-sensing: Recognising cyclists' route, route direction, and route progress from cell-phone audio. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 453–457. IEEE, 2013.

[16] D. Weinland, R. Ronfard, and E. Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224–241, 2011.

[17] C. Zhu and W. Sheng. Human daily activity recognition in robot-assisted living using multi-sensor fusion. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 2154–2159. IEEE, 2009.