

iLearn on the iPhone: Real-Time Human Activity Classification on Commodity Mobile Phones

T. Scott Saponas, Jonathan Lester, Jon Froehlich, James Fogarty, James Landay
University of Washington
Seattle, WA 98115
{ssaponas, jlester, jfroehli, jfogarty, landay}@cs.washington.edu

ABSTRACT

As computing moves beyond the desktop, human activity becomes an essential component of many applications. Activity classification is an active research area and several research systems have been constructed. Most have focused on fragile custom hardware only available in limited quantities. We instead seek to use commodity hardware to lower the barrier to creating activity-informed mobile applications. We describe *iLearn*, our system for classifying human activities using the Apple iPhone's three-axis accelerometer and the Nike+iPod Sport Kit. Our results suggest activities including running, walking, bicycling, and sitting can be recognized at accuracies of 97% without any training by an end-user.

Author Keywords

Activity Classification, Machine Learning, Mobile Phone

ACM Classification Keywords

H.5.2 [User Interfaces]: Input devices and strategies;

INTRODUCTION

Inferring human activities is an essential component of many current and future mobile-computing applications. Activity inference can, for example, enable models of human interruptibility, passive input for awareness applications, and direct input using locomotion to control an interactive application. Activity inference may also offer significant value to healthcare, eldercare, and fitness applications by providing doctors, family members, and users with a better understanding of daily physical activities.

Mobile activity classification efforts for these purposes have often focused on the construction of custom research hardware devices [1, 2, 6, 7]. However, such custom hardware is often not appropriate nor designed for long-term deployment with users outside of the laboratory. These sensor platforms are typically expensive to build, difficult to maintain, and often cumbersome or unattractive for users to wear (e.g., a sensor around the belt [2] or hanging from the neck [13]). Because it is difficult to manufacture, distribute, and maintain custom hardware on a large scale, these sensor platforms are generally not widely available within the research community (though, efforts like the MSP Research Initiative [2] should be recognized). Research focused on user studies of activity inference in applications is often stymied by this lack of adequate resources and the difficulties associated with custom

hardware. Utilizing commodity devices for activity inference provides researchers with access to robust, readily-available hardware and potentially large preexisting user bases. It therefore is an attractive and important step toward everyday activity-informed applications on mobile phones.

We seek to lower the barrier to creating activity-informed applications on mobile phones by demonstrating how the iPhone can be used to create applications with real-time activity classification using the iPhone's three-axis accelerometer and the Nike+iPod Sport Kit. The iPhone is an attractive device for these applications because of its rich multitouch user interface, location sensing framework, fast processor, and highly available network connection. We augment these capabilities by demonstrating the possibility of very accurate classification of simple human activities such as running, walking, bicycling, and sitting. We also provide a set of open source tools for collecting training data, building activity models (using the Weka toolkit [12]), and running these models on the iPhone to provide real-time activity classification to applications.

The remainder of this paper is organized as follows. First, we briefly review related work in mobile human activity classification. Next, we present our activity classification platform. Later, we describe an offline analysis of data we collected to explore the feasibility of our approach to human activity classification. We also show how *iLearn* can be used to build an interactive game using human motion. Finally, we conclude with a discussion of possibilities for more activity-informed applications in future work.

RELATED WORK

There have been several custom systems built to detect activities in real-time. The MSP [2] is a belt worn sensing platform that includes a 3-axis accelerometer, barometric pressure sensor, microphone, and light sensors. These sensors are used to classify, in real-time, activities such as walking, standing, jogging, riding elevators, and stair climbing. The eWatch [7] is a custom made wrist-watch form-factor system that was built to detect similar activities.

There are also several sensor networking systems that either perform inference offline or use laptops (or other infrastructure) to perform computation. For example, Tapia *et al.* developed a system that uses five wireless

accelerometers, a heart rate monitor, and laptop to track similar activities in real-time [14].

Several companies have released products equipped with sensors designed to track fitness activities, such as the Nokia 5500 Sport [9] and the Nike+iPod Sport Kit [10]. However, these commercial products offer limited value to researchers in that they detect a very small set of activities (e.g., walking) and are designed to be closed platforms.

PLATFORM FOR ACTIVITY CLASSIFICATION

In this section, we briefly describe our platform for human activity classification, the tools we developed to classify activities, and the features we extract from the iPhone's accelerometer and Nike+iPod sensors.

To guide the construction of our platform, we began by collecting pilot data from three members of our research group performing a set of the activities we were interested in classifying. We analyzed this data to help determine an appropriate machine learning algorithm to use, select candidate features for classification, and construct our tools.

iLearn Platform for Mobile Activity Classification

Our platform consists of tools for gathering labeled data, learning and testing models with labeled data, and for real-time activity classification in iPhone applications using learned models.

iLog: Mobile Phone Tool for Gathering Labeled Data

iLog is a tool we created for collecting labeled training data on the iPhone for later use in machine learning and classification. When iLog is run on the iPhone, iconic radio buttons are displayed for each available activity. To record an activity, the user selects the activity's radio button, presses record, and then begins performing that activity. Once finished, the user presses stop. While iLog is recording, it generates features from the sensor data in real-time and saves those features to disk, along with the currently selected label and the raw sensor data.

iModel: Desktop Tool For Learning and Testing Models

Data saved by iLog can be imported into iModel, our Java application built on the Weka machine learning toolkit. In iModel, labeled data can be used to learn a model or test an existing model. For example, iModel supports loading up multiple files for a hold-one-out test, where each file is tested with a model built from all the other data files. Hold-one-out tests are useful for validating that models built with data from several people correctly classify data from a new person.

iClassify: Library for Real-Time Classification in iPhone Apps

Once a model has been learned using iModel, it can be saved to file and loaded into our iClassify library on the iPhone for *real-time* activity classification. iClassify can be used in iPhone applications written in Objective-C. The API is simple: applications specify the model file and register for a callback. iClassify then starts reporting activity classifications approximately once per second.

Features

We employ 124 features over the iPhone's 3-axis accelerometer and the Nike+iPod Sport Kit accessory. The accelerometer is sampled at 200 Hz and the Nike+iPod sensor in the shoe transmits data packets approximately once per second to the iPhone.

Nike+iPod Packet Payload Features

The Nike+iPod Sport Kit is an add-on for the iPod Nano enabling users to track their running pace and distance during a workout. It consists of a sensor placed in the shoe and a receiver plugged into the iPod. The sensor wirelessly transmits running information to the receiver.

While the Nike+iPod Sport Kit is not an officially supported accessory for the iPhone (Apple only provides the accompanying running software for the iPod Nano), we have been able to configure the iPhone to listen for nearby shoe sensors using the built-in serial port and the Nike+iPod receiver. There is no hardware modification involved; the receiver is simply plugged into the iPhone and the user must dismiss the "This accessory is not supported by iPod" alert that is displayed by the iPhone.

The format of the data in the payload of Nike+iPod sensor packets is undocumented. We have not been able to determine any obvious format to the data and instead simply use the individual bytes from these packets directly as numerical features. While the content of the payload is unknown, the shoe sensor only actively sends data when the user is involved in a walking or running like movement and otherwise 'sleeps'. When the shoe sensor is not broadcasting data we set all of the related features to 0. This creates an additional binary "foot in use" feature encoded into the packet payload.

Accelerometer Magnitude Features

Using the magnitude of each axis of the accelerometer, we create the following features over a one second window: the mean, standard deviation, minimum value, maximum value, min minus max, and max minus min.

Accelerometer Frequency Features

Approximately every second, we compute a 256 point Discrete Fourier Transform (DFT) over the last 1.25 seconds of samples over each axis of the accelerometer. From this DFT we generate one feature for the energy in each of the first 10 frequency components, a feature for the energy in each band of 10 frequency components, the value of the largest frequency component, and the index of the largest frequency component.

MODELING ACTIVITIES

We learn a model and classify activities over our extracted features using a Naïve Bayesian Network as implemented in the Weka machine-learning toolkit [12]. As a part of training, our numeric features are discretized using Fayyad and Irani's MDL method [4].

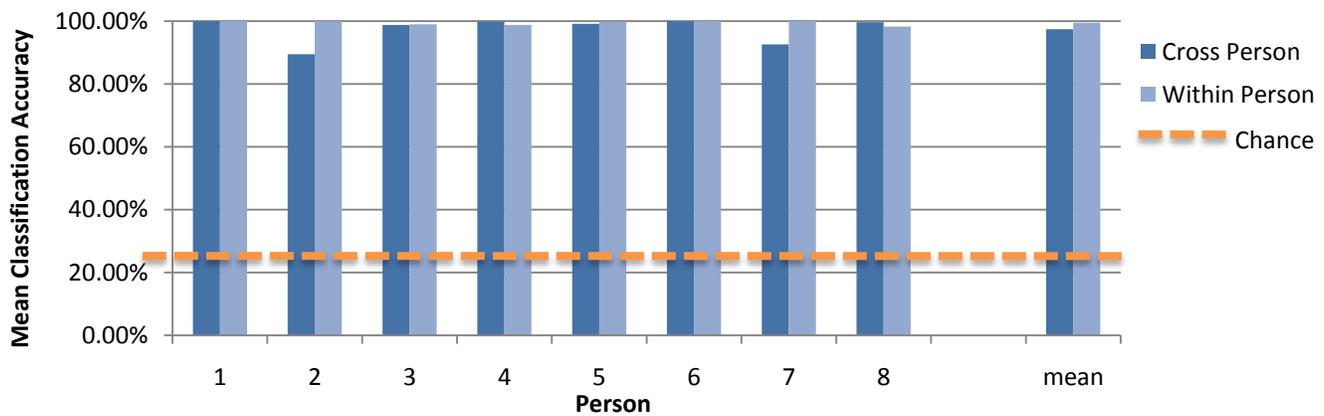


Figure 1. Classification accuracy results for offline hold-one-out experiments with cross-person and within-person models.

A Naïve Bayesian Network is a probabilistic classifier that models the probability of a class as independent probability distributions for each feature. Given an instance, the class is decided by applying Bayes' theorem across all features. We chose Naïve Bayes as our machine learning method because it was effective at classifying our pilot data and because classification using a trained model is computationally inexpensive. This property allows a potentially large number of applications on a mobile phone to each classify among a different set of activities simultaneously in real-time.

OFFLINE ANALYSIS

To explore the feasibility of our approach for classifying human activities, we collected data from eight computer science graduate students performing four activities useful for fitness related activity-informed applications: walking, jogging, bicycling on a stationary bike, and sitting. Note that these eight people and their data are distinct from the pilot data we collected while developing our system.

Our offline analysis dataset was collected after our system was finished being implemented. We collected approximately three minutes of data from each person. To balance the data for our analysis we found the user with the shortest duration, trimmed several seconds off the start/end (during which time the user would have been interacting with the interface on the phone), and selected this amount of data from each test person.

Data Collection

On two consecutive days each test person participated in a data collection session (two sessions total). In each session, they performed each activity once for three minutes. The order each person performed the activities in was different for their two sessions. The orderings were selected from two Latin Squares for each session, to avoid any ordering effects. They labeled the activity by pressing the corresponding button in iLog on the iPhone screen, and then put the iPhone in their pocket while they performed the activity. We did not constrain placement of the iPhone in their pocket or what types of pockets were worn. For some, the iPhone was in a tight pocket where the iPhone was in a fixed position. Others used loose pockets where the iPhone

and pocket could swing around as the person performed activities.

Results

We trained and tested classifiers using hold-one-out cross validation in two configurations: within-person models and cross-person models.

Within-Person Model

For each of the eight people, we built a model using their first session and tested on their second session. We also trained a classifier using the second session and tested it on the first. This yielded two folds for each of the eight people, or sixteen tests total. The mean accuracy across these tests was 99.48% (standard deviation 0.91%).

Cross-Person Model

We also built models based only on other people. For each of the eight people, we trained a classifier using data from the other seven people, testing it on their two sessions. As shown in Figure 1, the mean accuracy was 97.4% (standard deviation 4.05%).

Discussion

Within-person models achieved a mean classification accuracy of 99.48%, where random chance was 25%, suggesting that highly accurate classification is possible with only a small amount of training from the individual.

Cross-User models had a mean classification accuracy of 97.4%. This suggests that accurate activity classification might be achieved without the need to collect training data for new users. It also suggests our technique is somewhat robust to factors such as individual styles of movement and clothing.

MARIO FIT

In addition to a wide range of ambient and passive applications, iLearn can also support activity classification for direct input to interactive applications. As a demonstration of this potential, we modified NES.app [11], an open source Nintendo emulator for the iPhone, to bind traditional Nintendo controller input to human activities like running, walking, and jumping. The result is a mobile video game system, where a game like Super Mario Bros

can be played through physical activity. The user can, iPhone in hand, walk, run, and jump around their environment to control Mario. Training data can be collected by having people mimic the Mario character with their own physical motion while playing the game using the controller. Using collected data, a model can then be built, and activity classification can begin to replace the controller.

FACEBOOK REAL-TIME ACTIVITY STATUS

Mobile activity classification can also be useful for journaling or giving friends real-time awareness of your activities. As a simple example, we have built a Facebook application that can update or append to a user's status, as seen by their friends, based on their current activity.

FUTURE WORK

Our current system focuses primarily on fitness activities and as such could support many fitness applications. The breadth of potential applications can be increased by training in other environments or on other activities. We hope many people will use iLearn to create mobile applications experimenting with activity classification.

Location Stack

Many new types of activities and applications could immediately be supported by using the Core Location framework in the recently released iPhone SDK [5]. Movement information could be combined with classification of walking, running, and sitting to determine what mode of transit the user is currently engaged in (e.g., bus vs. driving a car). For example, our Facebook status application could be enhanced to provide location context such as "Biking Home" or "Running at the Park."

High Level Activities

Currently, our system only tries to recognize activities within each second of sensor data (e.g. was a person running in the previous second?). We think of these as *lower level activities*; they change when someone, for example, stops at a stoplight, intersection, or drinking fountain. *Higher level activities*, such as going for a run, biking to school, or having an intense workout, are not modeled in our system. We imagine applications building upon the output of iClassify to recognize such higher level activities.

Group Level Activities

As a mobile smartphone, the iPhone has access to a highly available GSM data network. Combining this with information such as location, it might be possible to build social-based systems to infer higher level group activities, enabling applications such as coordinating meet-ups with groups of friends, social network games, modeling social networks, and innovative ways of sharing experiences within one's social group.

CONCLUSION

We have created a system for classifying human activities in real-time on a commodity mobile phone. Our initial data collection suggests that models can be pre-trained and used

without any retraining by end-users. Using our iLearn platform, applications leveraging human activity classification can be created quickly and easily on a commodity device without creating or wearing custom hardware. We believe this opens the door to more widespread research on and adoption of activity inference in new mobile and ubiquitous computing applications.

ACKNOWLEDGMENTS

The authors would like to thank our study participants for helping us to collect our data.

REFERENCES

1. Bao, L., & Intille, S., Activity recognition from user-annotated acceleration data", In *Pervasive 2004*..
2. Choudhury, T., Borriello, G., Consolvo, S., Haehnel, D., Harrison, B., Hemingway, B., Hightower, J., Klasnja, P., Koscher, K., LaMarca, A., Lester, J., Landay, J., Legrand, L., Rahimi, A., Rea, A., & Wyatt, D., The Mobile Sensing Platform: An Embedded System for Capturing and Recognizing Human Activities, In *IEEE Pervasive Computing*, vol. 7, no. 2.
3. Consolvo, S., McDonald, D., Toscos, T., Chen, M., Froehlich, J., Harrison, B., Klasnja, P., LaMarca, A., LeGrand, L., Libby, R., Smith, I., & Landay, J.. Activity Sensing in the Wild: A Field Trial of UbiFit Garden, In *Proceedings of CHI 2008*, ACM Press.
4. Fayyad, U.M., and Irani, K.B., Multi-Interval Discretization of Continuous Valued Attributes for Classification Learning. Proceedings of IJCAI 1993.
5. iPhone Development. <http://developer.apple.com/iphone>
6. Kern, N., Schiele, B., & Schmidt, A., Multi-sensor activity context detection for wearable computing. In *EUSAI 2003*.
7. Rowe, A., Smailagic, A., & Siewiorek, D., Location and activity recognition using eWatch: A wearable sensor platform, In *Ambient Intelligence in Every Day Life*. Springer (2006).
8. MSP Research Challenge. <http://seattle.intel-research.net/MSP>
9. Nokia 5500 Sport. <http://europe.nokia.com/5500>
10. Apple – Nike + iPod. <http://www.apple.com/ipod/nike/>
11. NES.app: The Nintendo Emulator for iPhone. <http://www.zdziarski.com/projects/nesapp/>
12. Witten, I.H. & Frank, E., *Data Mining: Practical machine learning tools and techniques*, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.
13. Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., & Wood, K., SenseCam: a Retrospective Memory Aid", In *UbiComp 2006*.
14. Tapia, E., Intille, S., & Larson, K., Real-Time Recognition of Physical Activities and Their Intensities Using Wireless Accelerometers and a Heart Rate Monitor, In *ISWC 2007*.