Enabling Human Activity Recognition with Smartphone Sensors in a Mobile Environment

Chun-Ting Chen Wei-Po Lee

Abstract—**Human activity is a special kind of situation information that can be combined with different kinds of environmental data and then be used to determine appropriate service actions. In this study, we describe how we develop a smartphone-based mobile system that includes two core modules for recognizing human activities. Different machine learning methods with a feature selection scheme are adopted to perform activity recognition from the collected smartphone signals. A series of experiments are conducted to evaluate the performance of our activity-based framework. Moreover, we implement a mobile system on a smartphone-cloud platform to demonstrate that the proposed approach is practical and applicable to the real world applications.**

Index Terms—**activity recognition, smartphone sensors, machine learning, recommendation**

I. INTRODUCTION

In recent years, increasingly more sensors have been embedded into smartphones to track and collect various information about users and their environments. Human activity can provide a special kind of information and can be combined with the perceived environmental data to constitute some critical factors that are used to select appropriate service actions for users [1][2]. In this study, we develop a smartphone-based activity recognition system that infers human activities and the recognized results can be used to make appropriate event recommendations.

Human activity can be combined with the perceived environmental data to form a complete world state representation. Several steps are involved in human activity recognition (HAR): recording a user's behavior for a period of time, extracting the relevant information from the behavior sequences, and analyzing the sequence data to derive specific patterns. These patterns reveal how users prefer to consume the specific services, and the patterns can be used to make recommendations more suitable for them. To perform activity recognition for mobile users, different wearable sensor techniques have been proposed to work with computational methods. Among other intelligent products, smartphones increasingly become sophisticated due to the development of techniques and more new functions embedded to them. They are currently the most representative portable products in the evolution of information and communication technologies. As a result, these powerful state-of-the-art devices have attracted a great deal of research attention in the field of pervasive computing. These self-contained handheld devices have advanced features, such as mobile operating systems, broadband internet access and other computer-like processing capabilities, and these features make the devices highly suitable for activity recognition. Therefore, in this study we explore the use of smartphones as an alternative approach for identifying physical activities. The information obtained can be used as input for different types of service recommendation. Time series data collected from various ubiquitous sensors can be used to recognize user activities. However, the limited sensing ability and the intrinsic noisy character of the sensory data restrict the types of activities that can be identified.

The information extracted from smartphones is often utilized to develop location-based services with real-time estimated user locations. Researchers have also presented various supports on the device-side or server-side of a location-based framework to efficiently develop complete applications. For example, Kim *et al.* ([3]) proposed the development of smartphone-based systems to provide location information that could be combined with GPS and Wi-Fi positioning systems in order to generate user contexts.

Considering smartphone-based activity recognition, many approaches have been already presented in the literature. For example, Kwapisz's work exploited the triaxial accelerometer on a smartphone for HAR [4]. It was able to classify six locomotion activities over intervals of ten seconds. Some researchers performed signal Fourier analysis and a machine learning approach to predict activities of walking, running, cycling and driving by using the accelerometer data of a smartphone (e.g. [5][6]). In the same way, the work presented in [7] used a smartphone with embedded accelerometer for the classification of four activities. Research on HAR with smartphone mostly adopted accelerometers. This is because that these embedded inertial sensors were firstly introduced in the mobile phone market [8].

Based on different kinds of sensing data collected many techniques have been proposed to automatically recognize human activities [1][2]. They mainly rely on the supervised learning methods [9][10], and differ in the type and the number of the sensors, considered activities, adopted learning algorithms and other parameters. The training stage can be performed online or offline through the option of collecting live data from the users by following a predefined activity protocol [11]. Then the trained models are used online for the recognition of user activities.

To achieve our goal and solve the existing problems in activity recognition, some additional requirements are

C.-T. Chen and W.-P. Lee are with Department of Information Management, National Sun Yat-sn University, Kaohsiung 80424, Taiwan (e-mail: wplee@ mail.nsysu.edu.tw).

ISBN: 978-988-14047-4-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

needed to accomplish in developing an efficient activity recognition system in practice. The first is low overhead. During the process of extracting accelerometer data from mobile phones and sending the collected data to the server for further analysis, the overhead of operations inevitably causes some performance degradation. Therefore, it's imperative to keep the overhead as low as possible. The second is extensibility. Striving for excellence and high-performance code, the system needs to ensure that the addition of new functions and the modification of existing functions can be easily performed. The third is maintainability. It refers to an application that can be easily fixed and changed after it has been released. Also, it is important to select a good maintenance tactic, based on the past experiences.

In this work, we adopt machine learning algorithms to recognize a user's activities and event recommendation can be made accordingly. Comparisons are made of different methods for extracting the features of time and frequency domains from the smartphone signals. To realize the proposed framework, a mobile client and cloud server architecture is presented to reduce computational load and enhance data management. A series of experiments have been conducted to verify the proposed activity-based framework and to evaluate the recognition performance.

II. Smartphone-based Activity Recognition

A. System Framework

As mentioned above, we present a system framework for human activity recognition to help mobile users to access the most suitable services (e.g., making different types of recommendations). Our system framework has a cloud-based, client-server architecture. The client here means a smartphone responsible for collecting the user's activity information, recording his feedbacks, and performing the application services. The reason for using smartphones is because they are the most popular handheld mobile devices at present; other wearable devices can be included for collecting more information about users, depending on the availability. Meanwhile, the server is constructed on the cloud to manage user profiles, perform signals analysis, train classifiers for activity recognition, and carry out computation needed for recommendation.

To detect a specific user activity, the system needs to continuously collect sensor readings from his smartphone, and then extract features from these time sequence data to train classifiers for recognition afterwards. Based on the recognized results, the system uses various recommendation techniques to produce a candidate list for users. The users can activate the service to achieve the target tasks.

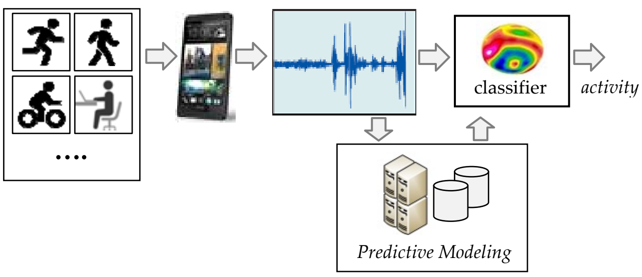

Fig. 1. The core units of the recognition system.

B. Sensors

In the activity recognition module, the embedded acceleration sensor is used for data collection. A smartphone has built-in triaxial accelerometers with 19.6 m/s2 maximum range. In general, the coordinate system of the smartphone is defined relative to the screen with its default orientation. As shown in Fig. 2, the horizontal scale is *x*-axis, with increasing positive values to the right; the vertical scale is *y*-axis, with increasing positive values upward; and the dorsoventral scale is *z*-axis, with increasing positive values outward. In this study, each sample of accelerometer data is sent over the network with the format of *x*, *y*, *z* values and a timestamp (in nanosecond).

The other sensor used in this work is GPS equipped in the smartphones. With this sensor, it is very convenient to develop context aware applications. The location information enables the trained model to make an accurate activity prediction. In this study, the GPS sensor is chosen to work as the location service provider, as it can provide accurate user information. In addition, most activities considered here are performed outdoors. Therefore, we do not address the GPS mapping problems that occur in the indoor applications. In the procedure of updating the location information, the relevant parameters are set to zero so that the users can receive the notifications as frequently as possible.

C. Feature Extraction and Activity Recognition

Activities are referred to as the ones occurring in a short time interval. They are brief and distinct body motions (such as walking) and can be characterized by a statistical sequence of body postures. Here, activities are constituted by readings of triaxial accelerometer that are collected within a short period of time, from a few seconds up to minutes. Six activities are considered, including standing, sitting, walking, jogging, biking, and climbing stairs. Smartphones with an open-source operating system are chosen as the operating platform here because they are easy to develop, deploy and maintain. To collect data, we create an interactive system to enable physical interaction between users and machines. The open-source smartphone system provides different ways (ranging from the lowest to the highest frequencies) for researchers to collect data from the accelerometer sensors, and we use the operating mode with a frequency of 50Hz for recording data.

Fig. 2. The direction of each axis on the smartphone.

The first phase in activity recognition is to extract specific target features from the time series data stream and then the system can infer what user activity is proceeding accordingly. In the feature extraction process, sensor signals are divided into a number of small time segments (called time windows) with the same time interval. Then, the signals are transformed into window-level feature dimension. The ideal length of the

ISBN: 978-988-14047-4-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

time window is determined by whether the specified interval provides sufficient time to capture several repetitions of motions involved in an activity. For each time window, features are derived from sensor data, and they are referred to as low-order features. In this study, data is divided into four-second segments and each segment contains 200 sensor readings. Each raw data (i.e., reading) is comprised of a timestamp, three acceleration values (corresponding to accelerations along the *x*-axis, *y*-axis, and *z*-axis), and two angle values (representing the latitude and longitude).

Two types of features are extracted: time domain and frequency domain features. According to the relevant studies, several time-domain features are used. They are the maximum acceleration for each axis, the minimum acceleration for each axis, the average acceleration for each axis, the average movement intensity (the average of the square roots of the sum of the values of each axis), the standard deviation for each axis, the mean absolute deviation (the average of the absolute values of sensor deviations), and the zero-cross rate (the number of times the signals change the sign in a given period of time).

In addition, to extract frequency domain features, we first convert the sensor data into the frequency domain vector, and then adopt the most popular frequency domain feature extraction, the Fast Fourier Transform (FFT), to derive features. The features are the average FFT spectrum for each axis, the FFT standard deviation of the spectrum data for each axis, the FFT energy (sum of the squared modulus of the coefficients) for each axis, and the FFT Entropy (the feature related to the amount of uncertainty about an event associated with a given probability distribution).

Considering both time domain and frequency domain, thirty-one informative features (categorized into eleven types) are generated from the 200 raw sensor readings in each time segment. These time and frequency domain features are used to constitute a feature vector for model training.

With the above two types of features, a selection scheme is performed to choose a subset of the original features to maximize the performance of a model-learning algorithm. In this way, the dimension of feature vectors required for activity recognition can be reduced, and the computational effort in learning a model (classifier) is thus decreased. In this study, the popular wrapper method for feature selection, Linear Forward Selection (LFS) is adopted to eliminate irrelevant and redundant attributes. LFS is an attribute selection method with a fixed-set technique for high-dimensional data. It can lower the number of attribute expansion in each forward selection step. More details are referred to [12].

After the features are selected, the data collected are transferred to the pre-defined feature vectors and several data modeling methods are employed to build the classifiers. The methods used in the experiments include Decision Tree, Multi-layer Perception, Random Forest and instance-based *k*-Nearest Neighbors. The performance evaluation and result comparison are presented in the experimental section.

III. Experiments and Results

A. Design and Implementation

A system called ActiFinder has been developed in a cloud-endpoint mobile environment to perform activity recognition, and the recognized results can be furthermore used to recommend specific events. As mentioned above, the system has a client/server architecture and is divided into two major parts: a user front-end (mobile client) and an application back-end (cloud server). It also includes a data storage module and a mechanism for client-server communication. The database contains tables for raw and intermediate data. Figure 3 presents the overall system design and the operational flow. As shown in the figure, the simple activities are recognized on the front-end, and the complicated computation (including the recognition of pre-defined complex activities) is performed on the back-end. In addition, the back-end includes an event filter that uses the recognized results to choose the suitable events to display, and a recommendation engine sending the event notifications to users.

The functions of the user front-end and the application back-end can be constructed in many ways, depending on their workloads and responsibilities. In this work, the user front-end is responsible for data collection, data transmission, and classification; while the application back-end is designed for classification, model generation, data storage, and data

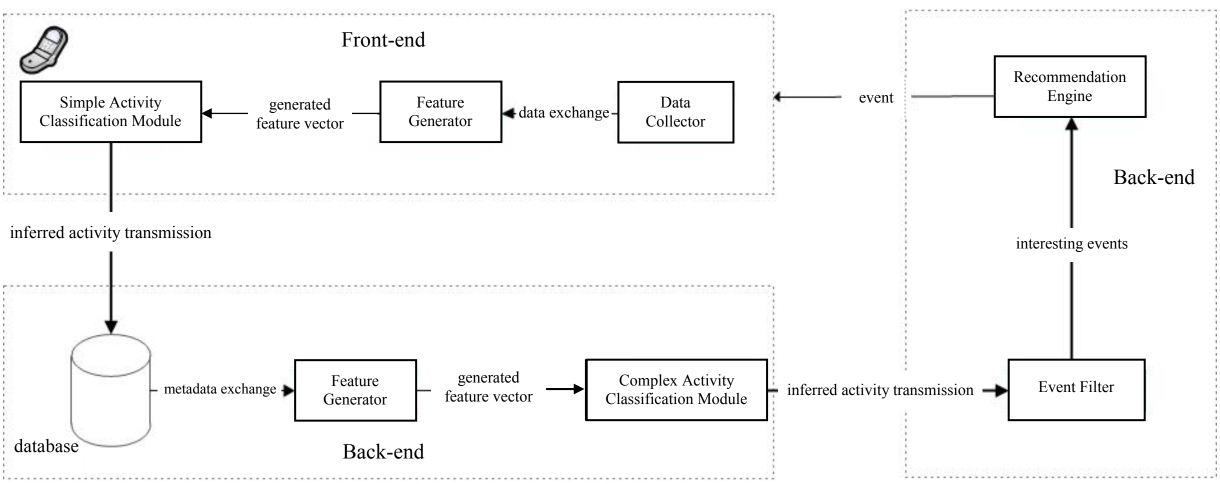

Fig. 3. The overall system design and operational flow.

ISBN: 978-988-14047-4-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

reporting. For the user front-end, the classification process occurs on the mobile phone using the pre-built IBK model because this model is efficient and it can deliver the best performance (according to the results shown below). In this way, the amount of computation required can be reduced. For the application back-end, the predictive modeling is performed on the server since it involves the complex data processing and other time-consuming steps. The tasks each part is responsible for are summarized in Table 1.

TABLE 1. Workloads for user front-end and application back-end.

Type	User Front-end	Application Back-end
Data Collection	✓	
Data Transmission	✓	
Classification	✓	✓
Model Generation		✓
Data Storage		✓
Data Reporting		✓

The user front-end has a graphical interface, where users can use the Android-based data collection platform to send relevant sensor data to the target server. Also, it provides the users with their current activity statuses and notifies the users the event information from their friends. Technically speaking, this app contains a trained model for activity recognition and reports interesting events from the server to users. On the other hand, the application back-end is literally a huge melting pot. It consists of an Apache HTTP Server, database, a model training module, an event filter and a recommendation engine. In this work, the Apache HTTP server is configured to run several PHP applications for the purpose of cross-language communication. In addition, the event filter decides which events should be displayed and the recommendation engine sends the activity notifications provided by the users' friends through a "like/request" option. The users can later express their preferences and intensions to engage in the same activity with their friends at that time and the place notified.

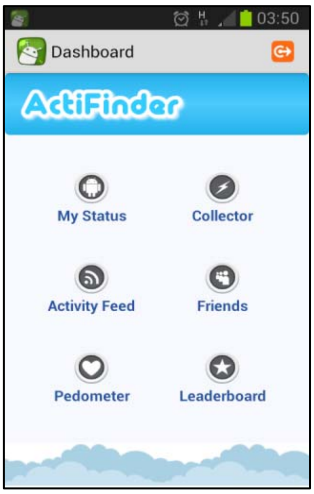

Fig. 4. Screenshot of the initial user interface

The activity recognition is performed at the user front-end, where the motion data are collected and the trained model is used to perform classification. Figure 4 shows the initial interface of our application system, through which a user can select which function he would like to consume. The interface for activity information is illustrated in Figure 5. The user can choose to collect and tag different types of activity data for training, and to view the data collected. Figure 6 is a typical example showing the event information from his friend and it is recommended to the user.

In this work, most of the data are stored in a MySQL database containing tables for raw and intermediate data. In this way, data visualization, methodology changes, and data insertion can be carried out conveniently. In addition, for each activity recognition task, data is stored in the respective table, and data from different users, times and activities can be mixed and merged to a single table. As inserting large amounts of data and retrieving the data back from the database often causes resource overhead and additional cost, the unavoidable degradation of system performance may occur. In order to reduce the overhead, the data processing mechanism is optimized, so that the system can save time waiting for the response from the database. In this work, a set of consecutive sensing records (i.e., 200 time steps) are compressed into a single sample and stored in ARFF format. This process dramatically reduces storage and communication bandwidth requirements. It is especially useful when the data need to be transmitted over the Internet.

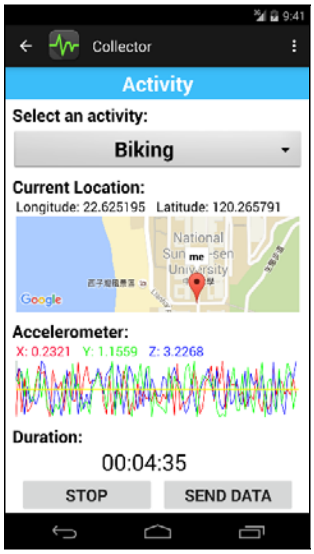

Fig. 5. Screenshot of the activity information

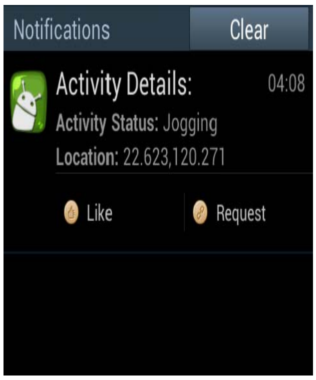

Fig. 6. Screenshot of social event recommendation.

In addition to the above functional modules, our work also includes a cross-language communication mechanism.

ISBN: 978-988-14047-4-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

The ultimate goal of cross language communication is to build a communication interface that communicates seamlessly across different programming languages such as PHP, JAVA, C++, .Net. The cross-language compatibility has become a trend of the next generation programming world.

Enabling an Android app to communicate with PHP, JSON (JavaScript Objection Notation.) is considered. It is a lightweight, human-readable, self-describing data-exchange format. The most important feature is that it is language and platform independent. Figure 7 shows the process of sending accelerometer and GPS data and retrieving the response.

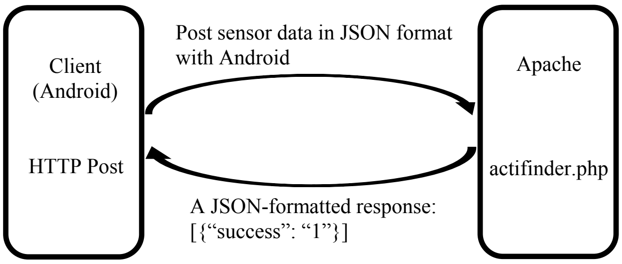

Fig. 7. The process of sending and retrieving data in JSON format.

B. Preliminary Results of Recognition

After developing the framework described in the above section, we conducted a series of experiments to evaluate the performance of four popular classification methods on six different activities. The methods include Decision Tree (J48), Multi-layer Perception (MLP), Random Forest (RF) and instance-based *k*-Nearest Neighbor (IBK). A set of participants were invited for data collection. Among the participants invited, nine of them completed the data collection. Each participant collected 51 minutes of sensor data, which were used to constitute 765 training instances (the time window was four seconds). In the experiments, the activity recognition was regarded as a multi-class classification task. The four methods mentioned above were used to build classification models to recognize the target activities. In addition, the ten-fold cross validation strategy was used for model testing. The results for the initial activity recognition experiments are summarized in Table 2. It presents the predictive accuracy (values were averaged over the participants) for each activity by the four learning methods. This table indicates that in most cases IBK achieves a relatively high accuracy.

TABLE 2. Results of different methods.

	Accuracy			
Activity	J48	MLP	RF	IBK
Jogging	63.4	84.4	82.4	83.8
Walking	71.4	82.3	70.1	80.4
Standing	66.3	68.6	78.7	88.8
Climbing Stairs	72.0	73.5	78.5	90.4
Biking	57.1	55.3	54.8	71.0
Sitting	80.8	82.7	85.3	94.6

As shown, some activities are apparently easier to be identified than others, depending on how the acceleration values of the sensor changed. In general, it is expected that the learned models are able to correctly classify different activities, as they involve different forward motion effects in acceleration. However, after examining the misclassified cases, we found that signals (data) of certain activities are similar with each other (such as jogging or biking), and this causes misclassifications.

IV. CONCLUSIONS AND FUTURE WORK

Human activity can be combined with other perceived environmental data to determine appropriate service actions. In this study, we developed a smartphone-based activity recognition system that inferred human activities to make appropriate event recommendation. In this work, machine learning algorithms were used to construct models to recognize user activities in different situations. Features of time and frequency domains from the smartphone signals were first extracted and then used with the learning methods to build predictive models. Results obtained from different methods were compared. In addition, to verify the presented activity-based approach, we implemented a mobile system on a cloud platform to demonstrate how the user activities can be tailored as situational information with mobile devices to develop a real-world application. We are currently investigating new ways for further system improvement, especially to overcome the difficulty of data collection.

REFERENCES

[1] D. J. Cook, N. C. Krishnan, and P. Rashidi, "Activity discovery and activity recognition: new partnership," *IEEE Transactions on Cybernetics*, 43, 820-828, 2013.

[2] O. D. Lara and M. A. Labrador, "A survey of human activity recognition using wearable sensors," *IEEE Communications Surveys and Tutorials*, 15, 1192-1209, 2013.

[3] Y. Kim, H. Shin, Y. Chon, and H. Cha, "Crowdsensing-based Wi-Fi radio map management using a lightweight site survey," *Computer Communications*, 60, 86-96, 2015.

[4] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," *SIGKDD Explorations Newsletter*, 12, 74-82, 2011.

[5] T. Plötz, N. Y. Hammerla, and P. Olivier, "Feature learning for activity recognition in ubiquitous computing," in: *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, pp.1729-1734, 2011.

[6] Z. He and L. Jin, "Activity recognition from acceleration data based on discrete Consine transform and SVM," in: *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pp. 5041-5044, 2009.

[7] M. Kose, O. D. Incel, and C. Ersoy, "Online human activity recognition on smart phones," in: *Proceedings of the Second International Workshop on Mobile Sensing*: *From Smartphones and Wearables to Big Data*, 2012.

[8] W. Z. Khan, Y. Xiang, M. Y. Aalsalem, and Q. Arshad, "Mobile phone sensing systems: a survey," *IEEE Communication Surveys Tutorials*, 15, 402-427, 2013.

[9] W.-Y. Deng, Q.-H. Zheng, and Z.-M. Wang, "Cross-person activity recognition using reduced kernel extreme learning machine," *Neural Networks*, 53, 1-7, 2014.

[10] Y. Liu, L. Nie, L. Han, L. Zhang, and D. S. Rosenblum, "Action2Activity: Recognizing complex activities from sensors," in: *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 1617-1623, 2015.

[11] O. D. Lara, A. J. Pérez, M. A. Labrador, and J. D. Posada, "Centinela: A human activity recognition system based on acceleration and vital sign data," *Pervasive and Mobile Computing*, 8, 717-729, 2012.

[12] M. Gutlein, E. Frank, M. Hall, and A. Karwath, "Large-scale attribute selection using wrappers," in: *Proceedings of IEEE Symposium on Computational Intelligence and Data Mining*, pp. 332-339. 2009.

ISBN: 978-988-14047-4-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)