GLOBAL JOURNAL OF ENGINEERING, DESIGN & TECHNOLOGY
(Published By: Global Institute for Research & Education)
www.gifre.org

ENERGY-EFFICIENT LOCATION EASY TRACKING WITH ANDROID MOBILE PHONE BY USING GPS

Rajamoses.R[1] & Sarooraj.R.B[2]
[1.]M.Tech(CSE) Student, SRM University, Ramapuram,Chennai,India.
[2.]Assistant Professor, Department of CSE, SRM University, Ramapuram,Chennai, India.

Abstract

Mobile data usage over cellular networks has been dramatically increasing over the past years. Smartphone is location based services using sensor set (GPS, WiFi, the acceleration sensor, the orientation sensor, etc.), consume more energy and continuously using GPS, can cause the complete battery drain within a few hours, Coverage areas of GPS are still limited that GPS typically cannot function indoors. EasyTrack, a location tracking service that leverages the sensor hints on the android smart phone to reduce the usage of GPS. Easy track executes a GPS sampling using the information from the acceleration and orientation sensors. Switches to the alternate location sensing method based on WiFi when users move indoors. A machine learning technique, Gaussian process regression used to reconstruct the trajectory from the recorded location samples. Easytrarck can significantly reduce the usage of GPS and still achieve a high tracking accuracy and provide storage, analysis and map visualization of routes of mobile users.

Keywords: *GPS, Trajectory, Path, Track, Compression, Sensor, and Android OS.*

I. Introduction

Nowadays, the number of users possessing and using smartphones equipped by a GPS receiver increases rapidly, while their interest for mobile applications that enable the storage, analysis and visualization of the collected space time information is apparent. In this paper, we present “EasyTracker” for Android O/S. The core of EasyTracker is threefold. At first, EasyTracker collects space-time points of user’s movement using a GPS receiver, thus creating and visualizing a path (or trajectory) followed by the user. In contrast to similar applications, the collected path, to be stored in a server’s database , is sanitized by filtering out locations that lie inside user-defined areas of interest – sensitive areas where the user does not allow to be tracked (e.g. around home, a hospital, etc.).“Easy Tracker” for Android O/S The core of it is threefold. At first, it collects space-time points of user’s movement using a GPS receiver, thus creating and visualizing a path (or trajectory) followed by the user. In contrast to similar applications, the collected path, to be stored in a server’s database, is sanitized by filtering out locations that lie inside user-defined areas of interest – sensitive areas where the user does not allow to be tracked (e.g. around home, a hospital, etc.). This way, it provides end users with personalized privacy functionality. The second key functionality is that EasyTracker allows a user to annotate parts of her track with labels and therefore to describe her current Such feature can turn out to be very useful for automatic fill-in of surveys performed in transportation science or for researchers working on activity recognition who are usually restricted to use manually processed surveys. Third, it encapsulates state-of-the-art algorithms that process in an online fashion the received stream of GPS recordings and transforms it into meaningful tracks, ready to be stored into a trajectory database for further analysis. Specifically, it includes line simplification methods that compress the incoming stream of time stamped locations (thus reducing the storage cost), which are then partitioned into homogeneous portions according to some spatio-temporal criteria using a state-of the- art segmentation method. According to this segmentation, the track is split into portions (i.e. sub-tracks), which can be labeled by tags that describe the corresponding spatio-temporal behavior of the user (e.g. STOPPED, when the speed is very low). This is important as it facilitates the user (or some auditing algorithm) to compare her manual annotations with the classified sub-tracks as provided by the segmentation algorithm. The big picture that illustrates these novel features of EasyTracker is depicted in Fig. 1.

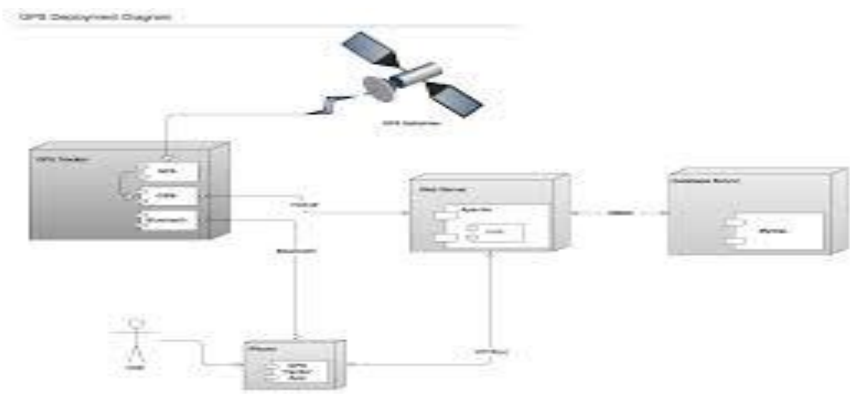

Figure1.System Architecture for EasyTracker

II. Motivation and Contribution

We motivate the present work by highlighting proposed system work evaluations. We demonstrate factors impacting energy efficiency in location sensing by employing mobile phones and summarize the limitations of existing smartphone usage that prevent energy-efficient location-sensing. We numerically analyze an optimal sensing interval for a smartphone in general open Wi-Fi networks and characterize key factors for energy-efficient Wi-Fi sensing. We present the WiFisense system that takes into account the factors identified in the analysis and provides mobility-aware Wi-Fi sensing capability for smartphones. We implement WiFisense on Android-based smartphones, and evaluate its performance extensively on both indoor and outdoor Wi-Fi testbeds.Via real-life experiments on our testbeds, we demonstrate that WiFisense reduces sensing frequency by up to 89%,while keeping a false-triggering rate low.

III. Related Work

In the general-purpose category, which targets to people who simply like to track and visualize their routes, Google's "MyTracks" is the most famous application for Android O/S. We have to note that this application is a relatively simple tracker with no advanced features, however it has more than 10 million downloads. Other similar applications in the same category include "AndAndo, "EveryTrail" , which has more advanced features by allowing users to add pictures or points of interest and create guides, and "OruxMaps", which allows utilizing various kinds of maps. Another category is about applications focusing on sports/fitness. Here we find many specialized features like the computation of calories consumption or the measurement of heart rate utilizing internal or external sensors.An applications designed to track lost devices or devices that belong to another person (assuming permission is granted). For instance, tracking a smartphone is feasible nowadays with the assistance of GPS and wireless networks.

Fig.2.Screen shot recording newtrack for EasyTracker

It used received signal strength changes from cellular towers as a trigger for Wi-Fi scanning. However, cellular signal strength varies considerably with changing locations and heterogeneous network environments (e.g., cell-tower density). In used a Bluetooth fingerprint to detect available APs. However, this requires a training phase to create and maintain the fingerprint database, which could be expensive given the fact that Bluetooth devices are usually highly mobile. In proposed to use a radio to detect Wi-Fi signals in ISM bands. However, this approach requires the use of a second radio, and thus might not be feasible for most smartphones. In propose to use habitual human mobility to forecast Wi-Fi connectivity. On the other hand, our work focuses more on energy-efficient discovery of Wi-Fi access opportunities.

IV. Literature Survey

A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone," in Proc. USENIX Conf. Annu. Tech. -Mobile consumer-electronics devices, especially phones, are powered from batteries which are limited in size and therefore capacity. This implies that managing energy well is paramount in such devices. Good energy management requires a good understand-ing of where and how the energy is used. To this end we present a detailed analysis of the power consumption of a recent mobile phone, the Openmoko Neo Freerunner. We measure not only overall system power, but the exact breakdown of power consumption by the device's main hardware components. We present this power breakdown for micro-benchmarks as well as for a number of realistic usage scenarios. These results are validated by over-all power measurements of two other devices: the HTC Dream and Google Nexus One. We develop a power model of the Freerunner device and analyses the energy usage and battery lifetime under a number of usage patterns. We discuss the significance of the power drawn by various components, and identify the most promising areas to focus on for further improvements of power management. We also analyze the energy impact of dynamic voltage and frequency scaling of the device's application processor. In this paper we attempt to answer this question and thus provide a basis for understanding and managing mobile-device energy consumption. Our approach is to measure the power consumption of a modern mobile de-vice, the Openmoko Neo Freerunner mobile phone, bro-ken down to the device's major subsystems, under a wide range of realistic usage scenarios.

K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao, "Energy-accuracy trade-off for continuous mobile device location," in Proc. 8th Int. Conf. Mobile Syst., Appl., Services, 2010 -Mobile applications often need location data, to update locally relevant information and adapt the device context. While most smart-phones do include a GPS receiver, its frequent use is restricted due to high battery drain. We design and prototype an adaptive location service for mobile devices, a-Loc, that helps reduce this battery drain. Our design is based on the observation that the required location accuracy varies with location, and hence lower energy and lower accuracy localization methods, such as those based on WiFi and cell-tower triangulation, can sometimes be used. Our method automatically determines the dynamic accuracy requirement for mobile search-based applications. As the user moves, both the accuracy requirements and the location sensor errors change. A-Loc continually tunes the energy expenditure to meet the changing accuracy requirements using the available sensors. A Bayesian estimation framework is used to model user location and sensor errors. Experiments are performed with Android G1 and AT&T Tilt phones, on paths that include outdoor and indoor locations, us-ing war-driving data from Google and Microsoft. The experiments show that a-Loc not only provides significant energy savings, but also improves the accuracy achieved, because it uses multiple sensors. Our goal is to develop location as a system service that automatically manages location sensor availability, accuracy, and energy. From an application developer perspective, this simplifies the use of the multiple existing, and potentially forthcoming, location technologies with varying characteristics. From a mobile user experience perspective, this allows the system to optimize battery life by intelligently managing the location energy and accuracy trade-offs based on available sensor capabilities. This is beneficial for mo-bile platforms that allow several third party applications to run on the platform, but at the same time must ensure long battery life for acceptable user experience.

Z. Zhuang, K. Kim, and J. Singh, “Improving energy efficiency of location sensing on smartphones,” in Proc. 8th Int. Conf. Mobile Syst., Appl., Services, 2010-Location-based applications have become increasingly popular on smartphones over the past years. The active use of these applications can however cause device battery drain owing to their power-intensive location-sensing operations. This paper presents an adaptive location-sensing framework that significantly improves the energy efficiency of smartphones running location-based applications. The underlying design principles of the proposed frameworkin-volve substitution, suppression, piggybacking, and adaptation of applications’ location-sensing requests to conserve energy. We implement these design principles on Android-based smartphones as a middleware. Our evaluation results show that the design principles reduce the usage of the power-intensive GPS (Global Positioning System) by up to 98% and improve battery life by up to 75%. In this paper, we present an energy-efficient location-sensing framework that effectively conserves energy for smartphones running LBAs. In its core, the proposed framework includes four design principles: Substitution, Suppression, Piggybacking and Adaptation. Briefly, Substitution makes use of alternative location-sensing mechanisms (e.g., network-based location sensing) that consumes lower power than GPS. Suppression uses less power-intensive sensors such as an accelerometer to suppress unnecessary GPS sensing when the user is in static state. Piggy backing synchronizes the location sensing requests from multiple running LBAs. Adaptation aggressively adjusts system-wide sensing parameters such as time and distance, when battery level is low.

J. Paek, J. Kim, and R. Govindan, “Energy-efficient rate-adaptive GPS-based positioning for smartphones,” in Proc. 8th Int. Conf. Mobile Syst., Appl., Services, 2010-Many emerging smartphone applications require position information to provide location-based or context-aware services. In these applications, GPS is often preferred over its alternatives such as GSM/WiFi based positioning systems because it is known to be more accurate. However, GPS is extremely power hungry. Hence a common approach is to periodically duty-cycle GPS. However, GPS duty-cycling trades-off positioning accuracy for lower energy. A key requirement for such applications, then, is a positioning system that provides accurate position information while spending minimal energy. In this paper, we present RAPS, rate-adaptive positioning sys-tem for smartphone applications. It is based on the observation that GPS is generally less accurate in urban areas, so it suffices to turn on GPS only as often as necessary to achieve this accuracy. RAPS uses a collection of techniques to cleverly determine when to turn on GPS. It uses the location-time history of the user to estimate user velocity and adaptively turn on GPS only if the estimated uncertainty in position exceeds the accuracy threshold. It also efficiently estimates user movement using a duty-cycled accelerometer, and utilizes Bluetooth communication to reduce position uncertainty among neighboring devices. Finally, it employs cell tower-RSS blacklisting to detect GPS unavailability (e.g., in-doors) and avoid turning on GPS in these cases. We evaluate RAPS through real-world experiments using a prototype implementation on a modern smartphone and show that it can increase phone life-times by more than a factor of 3.8 over an approach where GPS is always on.

Z. Zhuang, K. Kim, and J. Singh, “Improving energy efficiency of location sensing on smartphones,” in Proc. 8th Int. Conf. Mobile Syst., ppl., Services, 2010-The constrained battery power of mobile devices poses a serious impact on user experience. As an increasingly prevalent type of applications in mobile cloud environments, location-based applications (LBAs) present some inherent limitations concerning energy. For example, the Global Positioning System based positioning mechanism is well-known for its extremely power-hungry attribute. Due t o t he severity of the issue, considerable researches have focused on energy-efficient locating sensing mechanism in the last a few years. In this paper, we provide a comprehensive survey of recent work on low-power design of LBAs. An overview of LBAs and different locating sensing technologies used today are introduced. Methods for energy saving with existing locating technologies are investigated. Reductions of location updating queries and simplifications of trajectory data are also mentioned. Moreover, we discuss cloud-based schemes in detail which try t o develop new energy-efficient locating technologies by leveraging the cloud capabilities of storage, computation and sharing. Finally, we conclude the survey and discuss the future research directions.

V. Existing System

The localization technologies used mainly based on Global Positioning System (GPS), other technologies also obtain assistance from WiFi and GSM, each of which can vary widely in energy consumption and localization accuracy. As it is known to be more accurate, GPS is often preferred on mobile platforms over its alternatives such as GSM/WiFi based positioning systems. User still needs to connect GPS using internet connection active where ever user moves on different location, and verify the GPS signal. There is no point to start tracker storing the coordinates. Whether a location is stored is the decision of the compression.

VI. Proposed System

The proposed system works based on the recording a track is a few steps are required to record a track. As it is illustrated in Figure, the application at first verifies the GPS signal. If it is available, the tracker starts storing the coordinates. Whether a location is stored is the decision of the compression method enabled by the user. Each time a set of coordinates are stored into the database, the application, Uses the TSA(Temporal & Spatial algorithm) to segment, if necessary, the track and calculates various useful statistics (total distance of the track, average speed, etc.). We decided to store information in a local database in order to avoid requiring Internet connection. For this purpose, we make use of the Sqlite data base which is embedded on the Android SDK . The database is a simple schema that contains the main information of a track, the GPS coordinates it derives from, as well as the statistics described above. Accompanying places (POIs) and pictures are also stored in the database.

VII. Conclusion and Future Work

In this paper, we presented EasyTracker, a mobile application developed for the Android O/S that enables the storage, analysis and map visualization of routes of mobile users. In comparison with related applications, it provides novel functionality at three levels: (i) it enables users to manually annotate part of their routes with labels describing their activity and behavior, (ii) it encapsulates several state-of-the-art trajectory compression algorithms for a tradeoff between storage cost and quality of movement's representation, and (iii) it automatically segments tracks according to a state-of-the-art trajectory segmentation algorithm in order to facilitate automatic auditing of the user's manual annotation. As a fourth (preliminary) contribution, it enables users to protect their privacy by defining sensitive areas where recording is not permitted. This type of applications can be used in several fields, from route planning and resources administration (e.g. carpooling) to entertainment and social networking (e.g. next generation location-based social networks). Challenges for future work include the extension of TSA algorithm in order to recognize for each track segment the activities of the user in such a way as to enable the automatic validation and auditing of the user's annotation, an energy saving policy since our application is used in devices with limited energy resources, a possibility to detect user's position in non GPS-available areas (e.g. in indoors environments or in subsurface regions) without losing in accuracy and a hybrid local – in the cloud storage schema that would make it ready for social networking applications and can make use for Location based alarm where to user wants to reach out. A thorough understanding of the Internet requires detailed information about its topology and its performance. Given the size of the Internet, an approach based on crowd-sourcing may fit the scope in an unprecedented way: the results of a large number of short-range measurements, carried out using currently available smartphones, can be combined together to generate a fine-grained map of the network. Besides their potentially huge number, the use of smartphones as network monitors provides other opportunities: i) performance is observed at the periphery of the network, where the majority of end-users is located, ii) mobility of terminals allows the monitoring system to collect dynamical and geo-referenced information.

References

[1] GPS Tracking Pro. URL: https://play.google.com/store/apps/details?id=com.fsp.android .

[2] Android developers, Location Manager. URL: http://developer.android.com/reference/android/location/LocationManager.html (accessed: 1 Jun. 2012).

[3] Android developers, SQLite Database. URL:http://developer.android.com/reference/android/database/sqlite/SQLiteDatabase.html (accessed: 1 Jun. 2012).

[4] Android developers, Supporting Multiple Screens. URL:http://developer.android.com/guide/practices/screens_support.html (accessed: 1 Jun. 2012).

[5] EasyTracker: An Android application for capturing mobility behavior Alexandros Doulamis, Nikos Pelekis and Yannis Theodoridis.

[6] SensTrack: Energy-Efficient Location Tracking With Smartphone Sensors Lei Zhang, Jiangchuan Liu, Hongbo Jiang, and Yong Guan.

[7].Z. Zhuang, K.-H. Kim, and J. Singh, "Improving Energy Efficiency of Location Sensing on Smartphones," in Proc. of ACM MobiSys, June 2010.

[8] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B.Krishnamachar, and N. Sadeh, "A Framework of Energy Efficient Mobile Sensing for Automatic User State Recognition," in Proc. of ACM MobiSys , June 2009.

[9]. Ananthanarayanan and I. Stoica, "Blue-Fi: Enhancing Wi-Fi Performance using Bluetooth Signals," in Proc. of ACM MobiSys , June 2009.

Biography

R.Rajamoses is a M.Tech (CSE) Student in the Department of Computer Science and Engineering, SRM University, Ramapuram Campus, Chennai, Tamil Nadu, India. He received Master of Computer Science (M.Sc) degree in 2000 from Bharathiar University, Coimbatore, Tamil Nadu, India. He research interests are Big Data, Cluster Computing and Android programming.

R.B.Sarooraj is an Assistant Professor in the Department of Computer Science and Engineering, SRM University, Ramapuram Campus, Chennai, Tamil Nadu, India. He received M.Tech (CSE) degree in 2008 .He has been professional teaching experience in 5 years. He research interests are Grid Computing and Cluster Computing, Android Applications, Data Warehousing and Software Engineering.