ISSN: 2278-0181

Energy Efficient Location Tracking by the Combination of GPS and WiFi Technology

Savita N Patil
Dept. Computer Science and Engineering
AMC Engineering College
Bangalore, India
Savitanp09@gmail.com

Bharathi R
Dept. Computer Science and Engineering
AMC Engineering College
Bangalore, India
Bharathi_saravanan@hotmail.com

***Abstract*--Mobile applications often need location data, to update locally relevant information and adapt the device context. While most smartphones do include a GPS receiver, its frequent use is restricted due to high battery drain. Although GPS is often preferred over its alternatives, the coverage areas of GPS are still limited (GPS typically cannot function indoors). To this end, our goal in this paper is to improve the energy-efficiency of traditional location tracking service as well as to expand its coverage areas. Here location tracking service that leverages the sensor hints from the smartphone to reduce the usage of GPS. It selectively executes a GPS sampling using the information from sensors and switches to the alternate location sensing method based on WiFi when users move indoors. Gaussian process regression, a machine learning technique, is then used to reconstruct the trajectory from the recorded location samples.**

Key words: Location tracking, smartphone, sensor.

I. INTRODUCTION

With the increasing pervasiveness of smartphones over the past years, several Location-Based Applications (LBAs) have been adopted by mobile users for always on contact for social-networking, businesses needs, and entertainment. Some instances of currently popular LBAs include mobile social networking, healthcare, local traffic and local restaurants.

In spite of the increase in processing power, feature-set, and sensing capabilities, the smartphones continue to suffer from battery life limitation, which hinders the active utilization of LBAs. Consumer and advertiser expenditure on location based services is very high 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.

Nowadays, as smartphones are capable to accomplish complicated tasks, we still face problems. The demand of computing and storage capability on mobile devices is rapidly increasing in recent years, whereas the battery manufacturing industry moves forward slowly.

Unfortunately, it is also well-known that GPS, the core enabler of many location based applications, is power hungry. The aggressive usage of GPS can cause the battery to completely drain within a few hours [3], [4]. Location based applications still cannot assume continuous and ubiquitous location access in their design because of the high energy expense for localization. Even within the limited hours of being activated, GPS may not function well all the time, especially when the mobile user is under the shelter of buildings due to the signal loss under indoor environment [5]. When GPS is unavailable, alternate location sensing techniques must be used to obtain the approximated location. The variability in accuracy provided by various location sensing technologies and the limits on their coverage areas pose additional challenges for application developers [6]. Thus accuracy would be provided by using multiple location sensors simultaneously.

In this paper, we present an energy-efficient location sensing framework that effectively conserves energy for smartphones running LBAs. A location tracking service that provides user's moving trajectory while reducing its impact on the devices' battery life. By applying different localization technologies like GPS, WiFi/GSM, we expand the coverage area compared to the traditional approach that only uses GPS. Adaptive sampling can be done by using the sensors hints from the smartphone. This LBA smartly selects the location sensing methods between WiFi and GPS, and reduces the sampling rate by utilizing the information from acceleration sensor and orientation sensor, two of the most common sensors found on smartphones today.

II. RELATED WORK

To track the users' locations, many energy efficient sensing approaches with adaptive sensing policies have been proposed to minimize the energy consumption [2]. With the objective of minimizing the location error for a given energy budget, EnLoc [2], an energy efficient localization framework, includes a heuristic with a local mobility tree to predict the next sensing time by utilizing the dynamic programming technique. Jigsaw [3] uses the information obtained from the acceleration sensor and the microphone to continuously monitor human activities and environmental context. According to the user's mobility patterns, a discrete time Markov Decision Process is employed to learn the optimal GPS duty cycle schedule with a given energy budget.

There are also works based on the observation that the required localization accuracy varies with locations. An adap-

ISSN: 2278-0181

tive location service for mobile devices, a-Loc [4] uses a Bayesian estimation framework to determine the dynamic accuracy requirement, and tunes the energy expenditure accordingly. It argued in [5] that given the less accuracy of GPS in urban areas, it suffices to turn on GPS adaptively to achieve this accuracy. The rate-adaptive positioning system for smartphone applications (RAPS) was then proposed to minimize energy consumption with given accuracy threshold by using the information of moving distance, space-time history, and cell tower-based blacklisting.

Smartphone's energy consumption has been a major concern in research for a long time, and a number of studies have been done to improve the energy efficiency of mobile devices. In order to understand where and how the energy is used, A. Carroll et al. [6] measured the power consumption of a modern mobile device (the Openmoko Neo Freerunner mobile phone), broken down to the devices major subsystems (CPU, memory, touchscreen, graphics hardware, audio, storage, and various networking interfaces), under a wide range of realistic usage scenarios.

In this paper we take efforts to achieve high energy efficiency by reducing the sampling rate of sensing users' locations. However, our work uses a novel approach by utilizing the acceleration sensors and the orientation sensors on smartphones to capture the geometric features of users' moving trajectories.

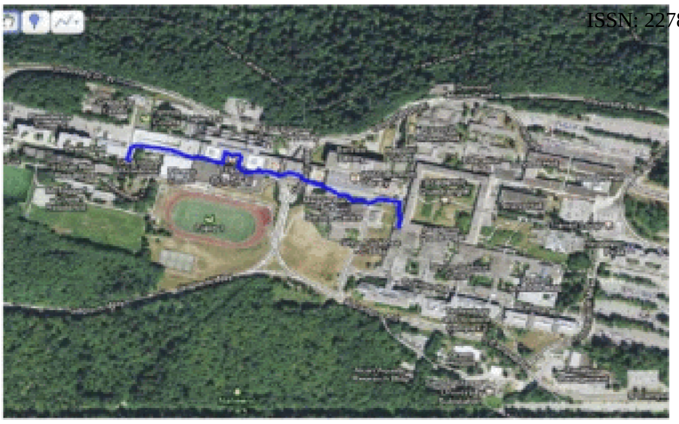
III. Challenges and Opportunities

In this section, we concentrating on the major defects such as limited availability and unnecessary samples of typical location based applications that utilize GPS. We then discuss the opportunities for making improvements.

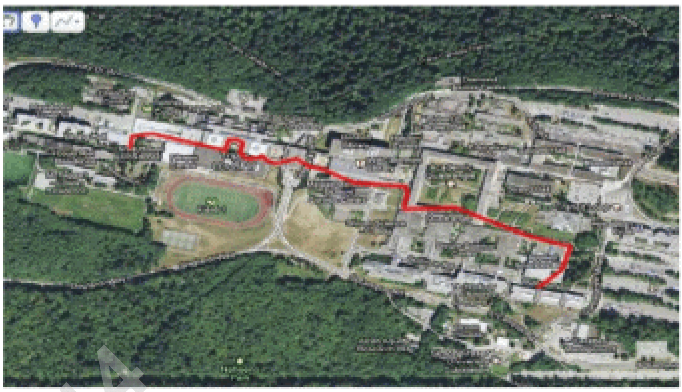
A. Localization using GPS and other location sensing methods

GPS cannot work properly under the indoor environment. The system provides continuous, world-wide, 3- dimensional position and velocity information to users with the appropriate receiving equipment.GPS also disseminates a form of Co-ordinated Universal Time (UTC). The satellite constellation nominally consists of 24 satellites arranged in 6 orbited planes with 4 satellites per plane. GPs can provide service to an unlimited number of users since the user receiver operate passively (i.e., receive only).

Figure 1(a) shows one track that we took using GPS on a mobile device. Although we did not stop recording, the track ends once it entered the building (the Academic Quadrangle in our campus), which indicates the performance of GPS largely depends on the working condition. The signals from GPS satellites can be blocked not only by buildings but also by canyon walls, trees, and even thick clouds. When the user walks through buildings, GPS equipped by a normal smartphone cannot function since the lack of satellite signals. Even worse, GPS units may consume more energy than the normal situation when there is no satellite signals [1]. Besides



(a)



(b)

Fig.1 Tracking results. (a) Track recorded by the naive approach. (b) Track reconstructed by our LBA.

GPS, there also exists alternate location sensing technologies. For example, Android OS provides a network based localization mechanism, which exploits GSM footprints from cell towers and WiFi signals to obtain an approximate location. Although the network-based location sensing is not as accurate as GPS, it provides the possibility to keep tracking inside a building since it mainly relies on the WiFi connection, in which case GPS units can be deactivated to save battery.

For the scenarios like university campus, hotels or hospitals, we can always assume persistent wireless local network access, which implies that other location sensing methods may provide us valid options when GPS is out of use.

B. Unnecessary GPS Samplings Versus Adaptive Sampling

The GPS sensor can sample the user's location at a relatively high rate. However, it is not ideal to record every location update since the error for each location sample varies. To make the path more smooth and fit the real trajectory, a typical location based application usually updates the user's location only if the distance to the last valid location sample is larger than a certain threshold [5]. Therefore, with a fixed and frequent GPS location sampling policy, it probably introduces a significant amount of unnecessary GPS samples. The system log of an Android application, My Tracks, which uses the GPS sensor in mobile devices to record the paths that users take

ISSN: 2278-0181

while hiking, cycling, running, or participating in other activities. The application usually takes several GPS samples to get one valid location update, in which case the threshold is 5 meters.

C. *Assistance From Other Sensors*

Nowadays smartphones become more and more powerful in terms of hardware, which usually contains various sensors. As an example, iPhone 4 is equipped with several environmental sensors, including an ambient light sensor, a magnetic compass, a proximity sensor, an accelerometer, and a three-axis gyroscope. Android 4.0 (API Level 14) also supports up to 13 kinds of sensors [8], even though the sensors' availability varies from device to device. The supported list of sensors in a Google Nexus S phone consists of: one KR3DM 3-axis Accelerometer, one AK8973 3-axis Magnetic field sensor, one AK8973 Orientation sensor, one GP2A Proximity sensor, one GP2A Light sensor, one Linear Acceleration Sensor, one Rotation Vector Sensor, one K3G Gyroscope sensor, and one Gravity Sensor [8].

To reduce unnecessary GPS samples, adaptive sampling is proposed in many existing works [3], [5]. Usually we need additional information to make adaptive sampling decisions, which may include the location history, the speed history, the distance information, remaining battery power, the accuracy requirement, etc. In this paper, we utilize the powerful sensors equipped by smartphones to obtain the information about changes of the orientation, moving speed, and traveled distance. Based on this useful information, we are able to make smart adaptive sampling decisions. The detailed design is described in the following section.

IV. System Overview

To reduce the frequency of location sensing, LBA periodically collects data from the corresponding sensor to detect a turning point or estimate current speed and the distance from the last recorded location.

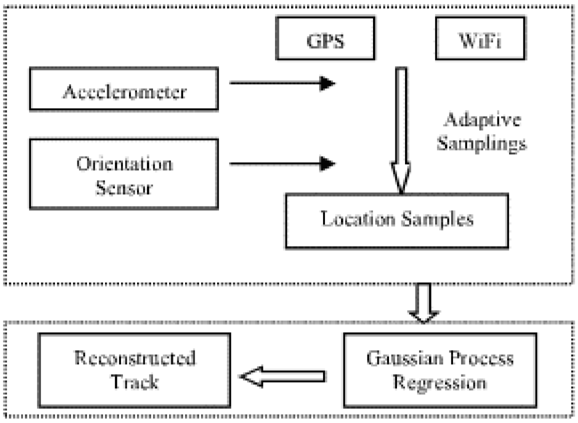


Fig. 2. The system architecture

The high energy efficiency of this approach is supported by the fact that the GPS sensor consumes much more energy than acceleration sensor and the orientation sensor [7]. When the GPS satellite signal is not available and the WiFi connection is active, LBA switches to the network based location sensing method to obtain the raw coordinates. The last step of this application is to upload the coordinates of sampled locations to an online server that uses a machine learning algorithm to reconstruct a smooth and accurate trajectory.

Figure 2 demonstrates the system architecture, consists of two stages: the first is to collect the location samples; and the second is to reconstruct the original trajectory. Given the working conditions, LBA switches between the GPS-based and the network-based localization methods using the GPS or WiFi sensors, respectively. By utilizing the sensor hints from the acceleration sensor and the orientation sensor, application is able to make smart adaptive sampling decisions in the GPS mode. For example, when the smartphone detects a turning point or if it estimates an unreasonable speed or an unexpected large traveling distance, it uses GPS to record the current location. After the server side receives all the collected location samples, a Gaussian Process Regression algorithm is then employed to predict the trajectory that the user has taken.

A. *Track Reconstruction: Gaussian Process Regression*

Once the collection of location samples is finished, it is not ideal to simply connect all the recorded locations, since the distances between any two successive locations may not be the same. For some parts of a trajectory, the recorded locations can be very sparse, while for other parts, the location samples may be relatively intensive. If we simply connect the location samples, the resultant trajectory can be very abstract. Therefore, uploading the collected data to the online server either by a wireless or wired connection to reconstruct the trajectory is our last stage.

.

TABLE I. Term Notations

x	Random variable
f(x)	target value of variable x
$f_*(x)$	predicted target value of variable x
F	vector of sampled data sets, and $f = [f_1, \ldots, f_n]$ on sampled data set X
f_*	vector for target value of the estimate data set
m(x)	mean function of random variable distribution, often considered as 0
k(x,x')	general form of covariance function
X	matrix of the sampled data set, where $X = [x_1, \ldots, x_n]$
X_*	Matrix of the estimate data set
Y	y is f(x) + noise, which is the real target value plus noise

We now briefly define GPR, a Gaussian process is a collection of random variables, any finite number of which has a joint Gaussian distribution, and is fully specified by a mean function and a covariance function. The inference of continuous values with a Gaussian process prior is known as Gaussian Process Regression. We first give some notations helpful for understanding, in the following table 4.1.

ISSN: 2278-0181

We adopt the Gaussian Process Regression (GPR), a machine learning technique to perform the interpolation. The training set of the algorithm is the recorded critical locations decided by the sensor hints which capture most of key features of a trajectory. And the testing set is the predicted locations between the successive but far-away location samples. Combing both input and output gives us the final trajectory. We next detailed describe GPR and how the user's trajectory can be reconstructed by using GPR. Consider x as a general random variable. We define the mean function m(x) and the covariance function k(x, x′) of a real process f(x) as

$$m(x) = E[f(x)],$$
$$k(x, x') = E[(f(x) - m(x))(f(x') - m(x'))],$$

and can write the Gaussian Process as

$$f(x) \sim gp(m(x), k(x, x')).$$

For notational simplicity the mean function is usually set to be zero. In our method the covariance function will be the squared exponential covariance function, although other covariance functions may also be useful. Assuming that observations are noise free, the covariance function specifies the covariance between the pairs of random variables

$$cov(f(x_p), f(x_q)) = k(x_p, x_q) = exp(- 1/2[x_p - x_q]^2) \qquad (1)$$

For a estimate data set X_* , we can generate a random Gaussian vector f_* for target values with the covariance matrix calculated from equation 1

$$f_* \sim N(0, K(X_*, X_*)),$$

Therefore, the joint distribution of the training outputs f and test outputs f_*. If X contains n training points, then $K(X, X_*)$ is the $n \times n_*$ matrix of the covariance evaluated at all pairs of training and test points. And the other entries $K(X, X)$, $K(X_*, X)$, and $K(X_*, X_*)$ are similar.

Algorithm 1 Predictions$(X, y, k, \sigma^2_n, x_*)$

1: $L = cholesky(K + \sigma^2_n I)$
2: $\alpha = L^T \backslash (L \backslash y)$
3: $f_* = k_* \alpha$
4: $v = L \backslash k_*$
5: $V[f_*] = k(x_*, x_*) - v^T v$
6: $\log p(y|X) = -1/2 y^T \alpha - \sum_I \log L_{ii} - n/2 \log 2\pi$
7: return $(f_*, V[f_*], \log p(y \backslash X))$

We then focus on explaining how to use GPR with given location samples to reconstruct the estimated trajectory. A trajectory can be considered as the path that the user fallows through space as a function of time. Specifically, we have n location samples from x_1 to x_n, each of which can be represented by a two dimensional points $x_i = (x_i, y_i)$.

B. Switching location sensing methods

Our LBA switches between GPS and the network based localization through the wireless connection. Here we use GPS for outdoors and the network based localization for indoors, and thus it is important to decide when to switch. Initially the application starts in the GPS mode and periodically executes a WiFi scan. When it detects the GPS signal loss as well as an active wireless network connection, application turns into the WiFi mode. If GPS become available again, and the phone loses the WiFi connection or the accuracy of location samples provided by the network decreases significantly, application switches back into the GPS mode.

There are two conditions to satisfy to switch the location sensing method: the current method fails to obtain location samples, and the other method is guaranteed to work, which prevents from switching between the two modes too often. Frequently changing location sensing mechanism can be very energy consuming, because the high-power components associated with both location providers need to be active. In some cases, both of the two methods are available when the user passing by some buildings. According to our rules, we should not change application's working mode, since in these situations the wireless connection tends to be unstable and short. In other cases, none of the two methods are available if we simply lose the GPS satellite signal outdoors. Our rules can also avoid the unnecessary switching in these cases. It is also worth mentioning that application stops collecting the sensor hints when it switches into the WiFi mode. In another word, we passively receive location updates in this mode. The reason is that, unlike GPS, when we request the location information, the WiFi localization technology cannot respond within a tolerable delay. It means that even if we apply the sensor hints to sense the location adaptively, we cannot obtain a location sample timely in the WiFi mode. Therefore, considering the WiFi localization updates the location less frequently than GPS, we decided not to waste energy on the acceleration sensor and the orientation sensor.

C. Utilizing the sensor hints

Acceleration sensor: It acts as a binary sensor to detect user movement or non-movement. The acceleration sensor in mobile device has been widely used in many existing location sensing systems. We do not limit the acceleration sensor just to be the user's movement detector, rather explore the possibility of calculating the distance that the user has traveled and also the speed.

Orientation sensor: orientation sensor as a detector of turning points when the user is moving. Application collects the readings of the orientation sensor, and computes the changes in direction. If user's moving direction changes dramatically (greater than the threshold θ), a location sensing of the user's current location is executed.

V. EVALUATION

We evaluated application using a real data set collected from a Google Nexus S phone carried by a mobile user walking in our university campus. The phone is equipped with an

ISSN: 2278-0181

integrated GPS, WiFi sensor, an accelerometer and an orientation sensor.

TABLE II. Average Error Predicted Locations

	Recorded locations	Predicted locations	Average error
Our LBA	38 samples	24 predictions	3.128m
GPS Trace	568 samples	0	0

We implemented our design prototype on Android 4.0 (API level 14). In modern mobile devices, the GPS receiver usually consumes much more power than the accelerometer and the digital compass. For example, our testing device, a Google Nexus S phone, is equipped with a BCM4751 integrated GPS receiver (produced by Broadcom), a KR3DM accelerometer (produced by STMicroelectronics), and an AK8973 3-axis electronic compass (produced by AsahiKasei Microdevices). With the battery supply (3.7 volt), the power consumption (in terms of current) of the accelerometer is 0.23 mA; and the current consumption of the compass is 6.8 mA; however, the current consumption of the GPS receiver can be as much as 80 mA.

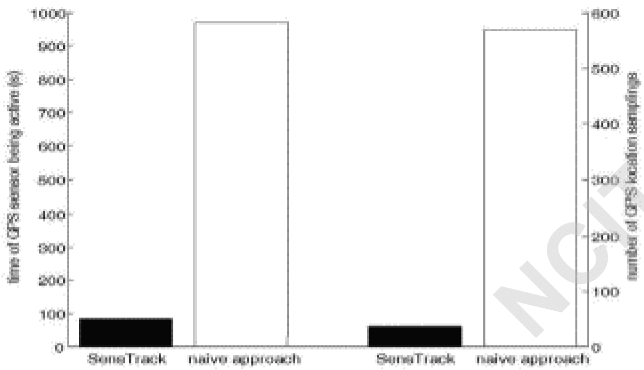


Fig. 3. Comparison of the energy efficiency

To demonstrate the energy efficiency of application, we present that it can significantly reduce the number of needed GPS samples and the time that the GPS sensor needs to be activated. For different hardware, the power consumption varies, and thus the energy consumption of LBA on a specific hardware model only provides limited information. Therefore, it is convincing and sufficient for us to show the relative energy efficiency of LBA to the naive approach by comparing the number of required sampling and the activated time of the GPS receiver. The result shown in Table II proves that LBA can achieve high accuracy. Figure 3 shows that compared to the naive approach, application only needs 7% GPS samples for the described path, and the time of the GPS sensor being active is decreased by nearly 90%.

CONCLUSION

In this paper, we have proposed a novel location tracking service. Our design can significantly reduce the usage of GPS and generate accurate tracking results by using trajectory reconstruction algorithm based on the Gaussian Process Regression. Although our work focuses on the pedestrians, it can be easily extended on multiple mobility patterns, such as running, biking, driving, etc, which are often with higher speeds. This is interesting challenges, which remain for future work.

REFERENCES

[1] C. Rasmussen, "Gaussian processes in machine learning," Adv. Lect Mach. Learn., vol. 14, no. 2, pp. 63–71, Apr. 2004.
[2] E. Kaplan and C. Hegarty, Understanding GPS: Principles and Applications. Norwood, MA, USA: Artech House, 2006.s
[3] O. Woodman and R. Harle, "Pedestrian localization for indoor environments," in Proc. 10th Int. Conf. Ubiquitous Comput., Sep. 2008.
[4] I. Constandache, S. Gaonkar, M. Sayler, R. Choudhury, and L. Cox, "EnLoc: Energy-efficient localization for mobile phones," in Proc. IEEE INFOCOM, Apr. 2009.
[5] S. Robinson, "Cellphone energy gap: Desperately seeking solutions," Strategy Anal., Newton, MA, USA, Tech. Rep. 4645, Mar. 2009.
[6] H. Lu, J. Yang, Z. Liu, N. Lane, T. Choudhury, and A. Campbell, "The Jigsaw continuous sensing engine for mobile phone applications," in Proc. 8th ACM Conf. Embedded Netw. Sensor Syst., 2010.
[7] 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.
[8] 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.
[9] 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.
[10] 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.
[11] M. Ra, J. Paek, A. Sharma, R. Govindan, M. Krieger, and M. Neely, "Energy-delay tradeoffs in smartphone applications," in Proc. 8th Int. Conf. Mobile Syst., Appl., Services, 2010.
[12] A.Carroll and G.Heiser, "An analysis of power consumption in a smartphone," in Proc. USENIX Conf. Annu. Tech. Conf, 2010.s
[13] M. Kjærgaard, J. Langdal, T. Godsk, and T. Toftkjær, "Entracked: Energy-efficient robust position tracking for mobile devices," in Proc. 7th Int. Conf. Mobile Syst., Appl., Services, 2011.
[14] L. Ma, J. Liu, L. Sun, and O. Karimi, "The trajectory exposure problem in location-aware mobile networking," in Proc. IEEE 8th Int. Conf. MASS, Oct. 2011.
[15] A. Cavanaugh, M. Lowe, D. Cyganski, and R. Duckworth, "WPI precision personnel location system: Rapid deployment antenna system and sensor fusion for 3D precision location," in Proc. Int. Tech. Meeting Inst. Navigat., Jan. 2010.
[16] M. Keally, G. Zhou, G. Xing, J. Wu, and A. Pyles, "PBN: Towards practical activity recognition using smartphone-based body sensor networks," in Proc. 9th ACM Conf. Embedded Netw. Sensor Syst., 2011.
[17] N. Patel, "The $10 b rule: Location, location, location," Strategy Anal., Newton, MA, USA, Tech. Rep. 6355, May 2011.
[18] R. Meng, J. Isenhower, C. Qin, and S. Nelakuditi, "Can smartphone sensors enhance kinect experience?" in Proc. 13th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2012.
[19] Android Developers Reference: Location Strategies [Online]. http://developer.android.com/guide/topics/location/strategies. html
[20] Android Analyzer Report [Online]. Available: http://android fragmentation.com/database/l/manufacturer/samsung/deviceModel/Nexu