Understanding Location Manager in Android and Implementing an Optimal Image Geotagging Application

Isha Sahu[1], Ishita Chakraborty[2]

[1] B.E, Department of CSE, Royal Group of Institutions, Affiliated to Gauhati University

[2]Asst. Professor, Department of CSE, Royal Group of Institutions, Affiliated to Gauhati University
Guwahati-35, Assam, India

***Abstract*— Keeping in consideration the high demand for mobile applications, this paper focuses on understanding and implementing location based services for aiding in day-to-day tasks. We have tried to design an image geotagging application for Android devices running version 2.2 or higher. Also a comparison is drawn based on accuracy of geo-spatial information between the native camera geotagging integrated with android phones and that of our application. The minimum time and minimum distance parameters for location updates are used in deciding optimum capturing points.**

***Keywords*— android, location-listener, image geo-tagging, GPS, minimum update interval.**

I. Introduction

Smartphones have revolutionised the generation of technology available in the market today. These hand-held devices contain many valuable information beyond the capacity of our imagination, from a simple contact number to images to audios and videos, to chat conversations and lately also the highly essential geographical position of the user, they have become indispensable. Using mobile cameras to easily capture a huge number of beautiful images is the most widely form of preserving memories. Only if we could also remember the exact location of the images captured, we can always recollect and relive the moments. We chose the Android platform because it is one of the fastest growing mobile operating systems on the market and targets a larger audience.

Currently there are many alternative applications that also allow tagging of images with location coordinates. Even android phones have the service of Global Positioning Satellites (GPS) to retrieve location co-ordinates in their native cameras. However, the major concern is the accuracy with which we are able to obtain the geo-spatial data. Most of the smartphones have high computing ability and come integrated with sensors like accelerometers, GPS receivers, light sensors, proximity sensors and offer a variety of network connectivity like the Global System for Mobile Communication (GSM), Wireless Local Area Network (WLAN), High Speed Downlink Packet Access (HSDPA) and Universal telecommunication system (UMTS). These services can be used to the maximum to develop efficient and innovative applications. Even though applications can undergo malfunctioning due to hardware issues, some connection dependent applications are affected by connectivity problems and data rates [1]. Also environmental conditions, mobility of users, available satellites and cell towers contribute in determining the connection's strength.

Android SDK allows us to use GPS and Network Location Provider to acquire the user location. GPS is mostly accurate but is limited to environment with less congestion and consumes more battery. Another alternative is to utilize the wireless signals and cell towers to obtain information optimally. In our automatic image geotagging system (PicStalk) we concentrate on the location data and suggest ways in which we can improve the accuracy for a better user experience.

The analysis has been conducted on Android phone (Samsung Galaxy Ace Duos) version 2.3 for both native camera and our implemented application. The main parameters considered for monitoring location updates were minimum time (MT) and minimum distance (MD).

II. Literature Survey

Earlier, handheld GPS receivers such as a Garmin Etrex, [2] were in use for a number of years and have been used to enhance student learning through geocaching and mobile mapping. With improvement, last decade has observed a new ability to geotag photographs by using two separate devices: a GPS receiver and a camera. These devices are linked by the time and date settings on each of the devices and then an external application was used to synchronize them.

Now smartphones have both digital cameras and assisted GPS in-built to the device. The relative ease and accessibility of geotagging has "generated a wave of geo-awareness" [3]. US News & World Report lists geotagging photos as one of "50 ways to improve your life in 2009" [4]. Friedland & Sommer [5] state that "all the major smartphone makers are now offering models allowing instantaneous upload of geotagged photos, videos, and even text messages to sites such Flickr, YouTube, and Twitter." In a recent international survey of geography and bioscience higher education practitioners conducted by the authors, geotagging was repeatedly cited as one area of technology that practitioners expect to see expanding over the next five years, a sentiment shared by Luo et al. [3] who suggest that, "with the availability of internet, GPS devices and smartphones, the proliferation and availability of geotagged media will continue to expand". Similarly, Johnson et al. (2010) [5] identify the use of mobile phones in education as one of the key areas in which they expect significant growth in the next 12 months.

In context with the accuracy, there have been suggestions in improving the Quality of Service (QoS) using crowdsourcing. [6]. Also comparative study with respect to accuracy of location data has been done between Android and iOS [7].

III. METHODOLOGY

Our motive was to establish a comparison between the GPS information provided by the native camera and a self-configured Location Manager (LM). As such our first target was to design an image geotagging system which will have the ability to take pictures, view the pictures, map view of user's current location and also attach the image to its location on map. The steps of implementation are discussed as follows.

A. Implementing the Camera

Android camera comes with an integrated GPS monitor which can be activated according to the will of the user. However, for a developer it is hard to control the GPS monitor already implemented in the camera. To access the camera hardware directly, we need to add the CAMERA permission to the application manifest. We used the Camera class to control camera settings and parameters. Since we are implementing our own camera, we will need to display a preview of what's being captured by the camera to allow users to compose their photos. After the picture is taken, we receive a byte array of the image and store it in appropriate format in the device's external memory. At the same time, we trigger a Location Listener (LL) from the application. This LL will be used to retrieve the current location of the user from where the picture is taken.

B. Measuring Location Data

Location data can be retrieved using up to three different methods- Satellite Positioning, Cellular Positioning and Wireless Positioning. Fig. 1 shows the different positioning systems [8].

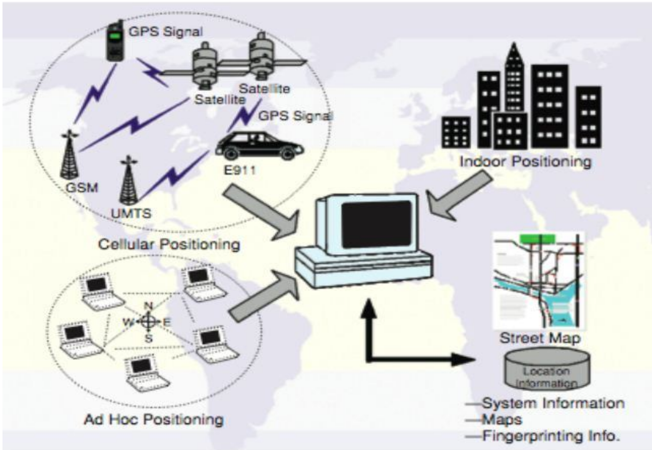

Fig.1 Positioning Systems

1) GPS: The most effective satellite positioning method is the GPS. It uses up to 32 satellites to determine the position of a Mobile Station (MS) [9]. To obtain accurate results, four or more satellites must be visible for GPS. In the case of a MS, when GPS is activated on the unit, the system uses triangulation to estimate the position of receiver when three or more satellites are connected. In case fewer satellites are available, then GPS receivers assume possibly degraded positions by reusing the last known location, dead reckoning, inertial navigation or other clues [10]. Although GPS is the most accurate positioning service but it is limited to being effective only outdoors and consumes lot of power on the end device. We need to keep in mind that our target is a mobile device with limited battery power.

2) Cellular Positioning: The GSM, UMTS generally referred to as 2G and 3G are the main cellular networks in operation. A cellular network comprises of a number of cells each containing a Base Station (BS) controlling all the MS. A cell is hexagonally shaped to provide equal signal strength in all area covered. A MS will search for the BS with highest signal strength to connect and that BS will be geographically closest to MS. The location of BS is already stored in database. So the approximate MS position can be retrieved using the BS location. The power consumption is relatively lower but the accuracy is compromised in this process because it depends on the size of the area covered by the cell tower [12]. Although cell towers can be used as location reference instead of GPS to reduce power consumption at the device [11], the accuracy of the location is relaxed.

3) Wifi Positioning: Here, the WiFi Access Points (AP) are kept track of in a database with geographical information. Most WiFi networks have an average of 60 meters so they provide good accuracy. No additional hardware is required and power consumption is lower than that of GPS.

With this we understand that GPS is most accurate but only suited for outdoor analysis; wireless, although mostly accurate is not available everywhere and cellular positioning lacks accuracy even though it is highly available and suited everywhere. Keeping these factors in mind we try to establish a LM combined with the cellular newtwork or WiFi (if available) for better results which is commonly referred to as Assited GPS (AGPS). In this prospect Ibrahim and Youssef have proposed a new algorithm based on a hidden Markov model which is especially suited for low end GSM phones and which delivers quite good results [13] and Nurmi et. al provided a grid based solution [14]. Also a probabilistic method based on RSSI fingerprinting is provided by Ibrahim and Youssef [15].

We obtain the geo-spatial data using the Application Package Interface (API) available. There are six types of location data that we can access. This data comes from a provider that can either be a cellular network or a GPS provider.

1) *Cached GPS*- Temporarily stores the last known GPS location which is required during first start of an application.

2) *Cached Network*- Temporarily stores the last known location of the cellular network provider. This information is gathered using cell network and WiFi.

3) *Real-time GPS*- Works only when GPS is turned on. Returns the location data and may take time to return value if used indoors. We can control the update time interval and distance i.e., MM and MD.

4) *Real-time Network*- Returns the information of location by cellular carrier. Here also, we can control the update time interval and distance i.e., MM and MD.

5) *Passive*- allows application to listen for location updates when in minimized state. Thus allows saving power consumption.

6) *NMEA*- Generally used in maritime apps and not in human readable form.

Before utilizing these parameters, we need to understand the requirement and scope of our application, its target audience and frequency of use. PicStalk will be mostly beneficial for students going on field trips or people going on tours or for those who never want to lose track of their image memories. They have an option between android native camera and PicStalk. Fig. 2 shows us the methods available with LM in android. We can alter the parameters and test accordingly the efficiency of the application in different scenarios. Later in this paper we will analyze which would be a better choice.

C. Storing Geo-data with Image

Android devices make use of Exchangeable image file format (Exif) that contains a set of predefined metadata tags. Any image captured from the device has an Exif attached to it.

```
locationManager.getLastKnownLocation(Location
Manager.GPS_PROVIDER);

locationManager.getLastKnownLocation(Location
Manager.NETWORK_PROVIDER);

locationManager.requestLocationUpdates(Location
Manager.GPS_PROVIDER,time,distance,listener);

locationManager.requestLocationUpdates(Location
Manager.NETWORK_PROVIDER,time,distance,list
ener);

<receiver
android:name=".PassiveLocationChangedReceiver"
android:enabled="true" />

addNmeaListener(GpsStatus.NmeaListener);
```

Fig. 2 Methods Available for Calling by a Location Manager in Android

This format is also recognizable by a variety of desktop applications like Picasa, Flickr which can extract information from the image. Android uses Exif to store the location data such as latitude, longitude, altitude with the image captured. A disadvantage of such file format is that even if the user is not willing to reveal information regarding his image, the format can be read and information can be extracted.

Keeping this in mind, we opted not to use Exif with PicSatlk images. Also, we create a hidden folder of all the images captured so that no other application can access the image or the information contained within it. We simply create a database of image that has columns to store values of latitude, longitude, timestamp, the image path in device and the reverse geocoded address. We have used SQLite, as Android is compatible with SQLite databases.

D. Displaying Images on Map

Using the external Maps library included as part of the Google API package, we can create map-based Activities using Google Maps as a user interface element. We have full access to the map, which enables us to control display settings, alter the zoom level, and pan to different locations. Using Overlays we can annotate maps and handle user inputs. Maps and location-based services use latitude and longitude to pinpoint geographic locations, but users are more likely to think in terms of a street address. The maps library includes a geocoder that we can use to convert back and forth between latitude or longitude values and real-world addresses. Map Views support annotation using Overlays and by pinning Views to geographical locations.

In our application, user has the option to view the image pinned to its captured location. The location of image is retrieved from the database. Fig. 3(a) and Fig. 3(b) show displaying of image on map using PicStalk and native android application.

Fig. 3(a) Image Map View using PicStalk

Fig. 3(b) Image map View using default Android Application

IV. Performance Evaluation

The performance evaluation was done based on the accuracy with which the user's current location is determined depending on two factors: the time taken and the error in distance. First of all, the current latitude (LA) and longitude (LG) of user was determined using Google Earth and locating manually where the user is. This value of LA and LG is then taken as the reference for rest of the calculations. We need to test both native camera application and PicStalk.

The following steps were performed for both the situations and corresponding readings were noted:

1) The default WiFi, GPS and GSM or any data service was disabled.
2) The camera was started.
3) For native camera, the GPS was enabled from camera settings. For PicStalk, application setting was used to enable GPS. For the first five minutes (300 seconds), readings were recorded by clicking different pictures within a range of 50 meters.
4) The WiFi, and GSM or any other data services available were activated.
5) Again readings were taken by capturing photos and monitoring the coordinates of LA and LG until a stable value is obtained for around fifteen minutes.
6) A table with LA and LG values for both the application was maintained for different time duration and the error in distance (in meters) was computed with the reference chosen earlier.

The following section will discuss the results obtained.

V. Experimentation and Results

A graph was plotted with time duration elapsed in seconds on X-axis and the distance error recorded in meters on Y-axis. Fig. 4 shows the plotted graph. From the graph we can evaluate that the minimum error in meters for android native camera is 150 meters, whereas that of PicStalk is 30 meters. Also, a picture was captured from both the applications and was displayed on map as shown in Fig. 3(a) and Fig. 3(b). As can be seen, PicStalk was able to pin the image on map with an accuracy of around 30 meters whereas the location of the same image was displayed with lesser accuracy using android default geo-tagging. Thus PicStalk application provides accuracy up to 30 meters if all the services are active for a MT of 35 minutes. Thus we obtain a better result than the native geo-tagging in android.

It is observed that it requires a MT of 35 minutes for location services to be fully functional and return a valid LA and LG value. Keeping all the services active will definitely drain out battery power faster. The targeted audience should not complain about battery consumption and at the same time should be satisfied with the LA and LG values their images are being pinned to.

To control this we need to optimize the LM implemented by establishing a tradeoff between the accuracy and power consumption. In order to do this, some tests were conducted to compare different MT update intervals: 8 seconds, 15 seconds, 30 seconds and 60 seconds.

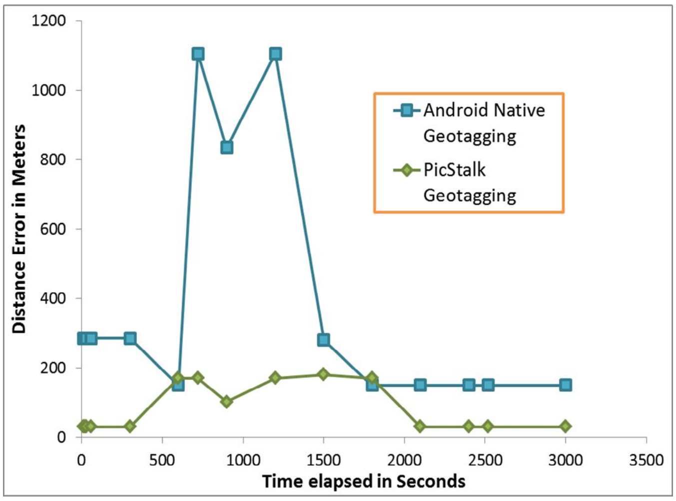

Fig. 4 Comparison of distance error between Android Native Geotagging and PicStalk

Fig. 5 depicts the location accuracy depending on GPS updating times. All the readings were taken with a maximum duration of 10 minutes. The values returned during that 10 minutes are compared with the referenced LA and LG. It is found that maximum accuracy of up to 49 meters occurs when the update interval is set to 8 seconds. Also highest battery consumption occurs during this update interval. The lowest accuracy of up to 136 meters occurss with 60 seconds of update interval which consumes the least battery power. The purpose of our work is to find the most accurate location with minimum power drain. So we choose the intermediate value of 30 seconds update interval. Thus, PicStalk allows GPS to update its location twice every minute without compromising much with the location coordinates.

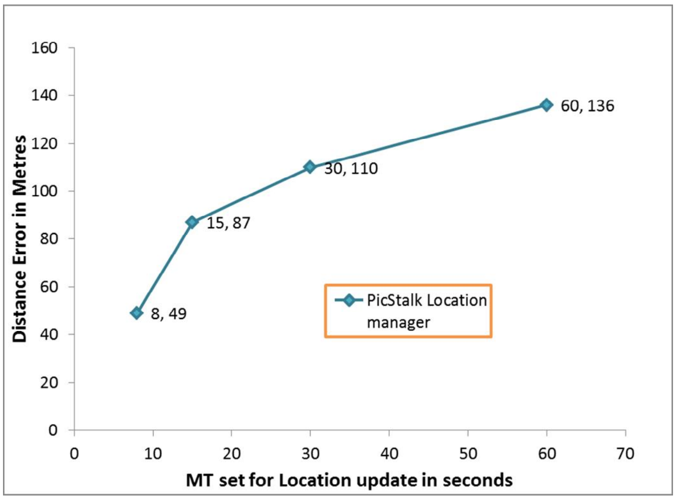

Fig. 5 Comparison of accuracy for different GPS update intervals

Thus, we see, users can opt for using additional applications for obtaining better results for image geotagging apart from the already existing native applications according to their needs.

VI. CONCLUSION

In this paper, we have tried to develop a consistent model for image geotagging, describing how the user's location data is obtained using Location Manager and Location Listener classes. Also the experimental results confirm that we were successfully able to draw a comparison between Android Native Geotagging and our application, Picstalk. With this, we understand that new applications can be designed to provide services as required by the audience. Moreover, for later studies, the proposed application can be modified to obtain a better trade-off between power and accuracy and more user friendly services can be added.

REFERENCES

[1] F. P. Tso, L. Zhang, L. Tengy, W. Jia, D. Xuany, and F. Zhang, "An empirical evaluation on the performance of mobile hspa networks." Technical report, Department of Computer Science, City University of Hong Kong, 2009.

[2] Fletcher, S., France, D., Moore, K. & Robinson, G., "Technology before pedagogy?" A GEES C&IT perspective, Planet, 5, pp. 52–55, 2003.

[3] Luo, J., Joshi, D., Yu, J., Gallagher, A. & Gallagher, A. "Geotagging in multimedia and computer vision: A survey, Multimedia Tools and Applications," 51(1), pp. 187–211, 2011.

[4] LaGesse, D.,"50 ways to improve your life: 'Geotag' your digital pictures", Available at http://www.usnews.com/articles/news/50-ways-improve-your-life/2008/12/18/geotag-your-digital-pictures.html, 2008.

[5] Johnson, L., Levine, A., Smith, R. & Stone, S. The 2010 Horizon Report (Austin: The New Media Consortium), 2010.

[6] Miquel martinez, David de Andres, Juan-carlos Ruiz, Mahbub Hassan, Salil Kanhere, "Towards chnaging the user perception of mobile communications through geotagged information," ARMOR '12, May 08 - 11 2012, Sibiu, Romania

[7] Denis Huber, "Background Positioning for Mobile Devices- Android vs. iPhone," Joint Conference of IEEE Computer & Communication Societies, 2011.

[8] G. Sun, J. Chen, W. Guo, and K. Liu, "Signal processing techniques in network-aided positioning: a survey of state-of-the-art positioning designs," Signal Processing Magazine, IEEE, vol. 22, no. 4, pp. 12 –23, July 2005.

[9] P. H. Dana., "Global positioning system Overview. [Online].Available:http://www.colorado.edu/geography/gcraft/notes/gps/gps f.html, December, 2010.

[10] GPS Support Notes (PDF). 19 January 2007. Archived from the original on 27 March 2009.

[11] I. D. Curcio, V. K. M. Vadakital, and M. M. Hannuksela, "Geo-predictive real-time media delivery in mobile environment," In 3rd Workshop on Mobile Video Delivery.

[12] U. Varshney, "Location management for mobile commerce applications in wireless internet environment," ACM Transactions on Internet Technology.

[13] M. Ibrahim and M. Youssef, "A hidden markov model for localization using low-end gsm cell phones," CoRR, vol. abs/1010.3411, 2010

[14] P. Nurmi, S. Bhattacharya, and J. Kukkonen, "A grid-based algorithm for on-device gsm positioning," in Proc. of the 12th ACM international conference on Ubiquitous computing, ser. Ubicomp '10. New York, NY, USA: ACM, 2010, pp. 227–236.

[15] M. Ibrahim and M. Youssef, "Cellsense: A probabilistic rssi-based gsm positioning system," Tech. Rep. arXiv:1004.3178, April 2010.