Smart Tourist - Passive Mobility Tracking Through Mobile Application

Sanjana Kadaba Viswanath[1], Chau Yuen[1(✉)], Xuefang Ku[2], and Xiang Liu[2]

[1] Singapore University of Technology and Design, Singapore, Singapore
yuenchau@sutd.edu.sg

[2] School of Software and Microelectronics, Peking University, Beijing, China

Abstract. Travel and tourism industry makes an evident impact on the world economy. Lack of real information on tourist sight-seeing places, is impacting trip planning for a major number of tourists. Growing smartphone users, increased sensing capabilities provided by smart phones, and the ability to share that data over the internet, presents an opportunity to build a mobile application that can provide an insight into mobility information through passive sensing. We have developed a mobile application to bridge the information gap by passively tracking tourist location and improvising tourist services using mobile crowd-sensing. Capabilities to cluster location data to form places of interest have been particularly focused in this paper. Finally, we present results for an initial experiment performed on a tourist to Australia.

1 Introduction

Tourism industry plays a big role in world economy. The Travel and Tourism industry contributes 2.2 trillion dollars to the world gross domestic product (9.5 %) and provides an impetus for creating 1 in 11 of the world's jobs, 4.4 % of total investment and 5.4 percent of total exports [1].

With the widespread adaptation of smartphones, with close to a billion being used by 2012 and expectation of 1.75 billion phones in 2014 [2], every major industry, including the tourism industry, is striving to adopt and expand on this platform to reach more customers and bridge the information gap. According to Trip Advisor's survey, 87 % percent of global travelers reported using a smartphone while on holiday [3]. The tourism industry has brought about sweeping changes to support this platform through review sites, booking websites, and information applications to improve experience for their customer.

Exponentially increasing smart phone users along with multiple sensing capabilities in-built in smart phones inspired us to build a mobile application to improvise tourism with simple yet powerful solution, providing a good opportunity through peer to peer sharing over passive tracking. Unlike the current solution that depends on the active review of the customer or paid reviews which

This research is supported and funded by Lee Li Ming Ageing Urbanism Programme, Lee Kuan Yew Centre for Innovative Cities, Singapore University of Technology and Design.

R. Giaffreda et al. (Eds.): IoT360 2014, Part II, LNICST 151, pp. 183–191, 2015.
DOI: 10.1007/978-3-319-19743-2_28

are sponsored by business owners, we focus on getting unbiased and passive feedbacks from real tourists.

Just like any Internet-of-Things (IoT) devices, smart phones provide computing, sensing and communicating capabilities and they fuel the evolution of mobility IoT as they feed sensor data to internet enabling either personal sensing or community sensing. To this end, mobile crowd sensing enables applications to continuously sense relevant sensor data for applications like intelligent transport systems [4], personal healthcare systems [5], etc. to process data to provide information. [6] demonstrates the effectiveness of mobile crowd-sensing to non-intrusively and passively collect mobile sensing data to detect wireless activity level in a large region.

In this paper, we propose a technology based on passive mobile tracking to provide an additional dimension to the review by using passive information about the trip for future travelers to fine-tune their trips and the tourism sector operators to get a better understanding of the places tourists spend their time in. We demonstrate a technology that bridges some of the trendiest developments in the current world's growth in tourism, widespread adaptation of smartphones, and use of mobile crowd-sensing to improve user experience. In this paper, we have detailed our current implementation by understanding the needs and requirements, the front end apps, the backend processing, and visualisation, and discussing results for an initial experiment on a tourist to Melbourne, Australia.

2 System Design Requirements

Tourists give a lot of importance for planning their trip and making sure the places they visit are worth the effort and time, so that they can make their holiday a memorable one. In a study for determining services for mobile tourist [7], results show that majority of tourists are interested in finding good accommodation, transport, food, and sight-seeing destinations. Although, there are many online review websites for hotel and food, we do not find any relevant information regarding sight-seeing destinations, especially those that provide us unbiased information on how much time other similar kind (eg., elderly-friendly, family-friendly, etc.) of tourists spare for a particular location. To this end, our objective was to provide recommendation for tourists by quantitative analysis on the places where other tourists visited during their trip, along with the time they spent in each place, and the route they took. This streamlines our system requirements as below:

- the place that they visited,
- how long they stay in each place,
- how people feel about the place,
- what that place is about (eg., food, hotel, sightseeing),
- if the place is family-friendly, elder-friendly, etc.,
- the activity the tourist is involved in, and
- how they access the place.

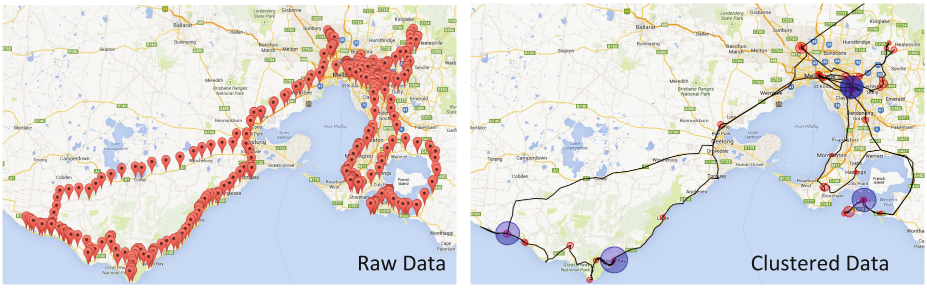

Fig. 1. Raw data(locations) vs clustered data(places)

In this paper, we describe a solution that passively tracks a tourist by fetching the locations that the tourist visited, and processing that as a place, and how long he stayed in each place. Figure 1, visualises data collected by smart tourist app. It compares raw data for the whole trip with processed/clustered data. It clearly shows clustered data shows useful information about the places visited by the tourist. Red circles are places of interest and blue circles are the places for accommodation. The path taken to go from one place to another is marked by black lines. The radius of the circles at each place denotes the amount of time the tourist has spent in that location. Our proposed solution not only provides recommendation to other tourists, but also provides a recorded memory of their trip through a personal diary on places they have visited without any effort.

Government agencies can benefit from our technology by planning their infrastructure to cater for the tourists needs, and improvise places that are less visited to invite more tourists. Our technology can aid businesses from getting a better knowledge of where tourists usually enjoy going, such that, they can have better business opportunities.

3 Mobile Application

In this section, we detail the design and implementation of various elements that have contributed to develop our mobile application. Location can be retrieved by GPS(Global Positioning System), Wifi Scan or Cellular network. We have used Android's Location API [8] to access the system location services to fetch location data.

3.1 Mobile Application Design

To encourage more tourists to learn from the feedback/experience from other tourists, we have developed an Android mobile application to fetch location data periodically from tourists' smart devices (eg., smartphones/tablets). The framework to fetch location data uses underlying Location API provided by Android SDK. After fetching location data, it is stored on local database on the device, which will later be uploaded periodically to cloud server.

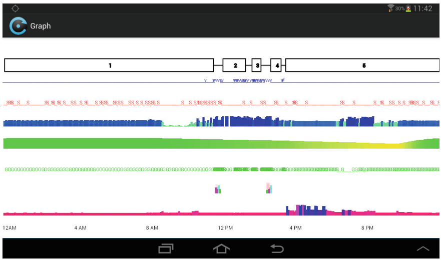

Fig. 2. Screen shot of mobile app collecting multiple sensor data

Location data is requested periodically, once every 5 mins by the background service that runs continuously and non-intrusively. GPS provides the most accurate location fix with accuracies up to 5 m, however, it is challenging to get location when GPS satellites are not in range. In which case, we fetch location from network or wifi-scan.

Figure 2, shows information of various sensor data collected for an entire day by mobile application. The first row represents clusters, whereas the second row is activity track such as walking (denoted as 'F'), on vehicle (denoted as 'v'), and bicycle (denoted as 'b'), or is stationary(denoted as '_'). The third row represents if the person is talking (denoted by 'S') or others (denoted by '_'). Forth row shows surrounding noise level and fifth row shows battery percentage. Sixth row shows how the location data has been retrieved, such as GPS (denoted as 'G'), or network (denoted as '_'). Seventh row shows speed information provided by location object, and lastly, eighth row shows light sensor values.

We use MySQL, Apache webserver, PHP, and RESTful HTTP to store and manage data at the server. For secure communications, we support only https calls to the server. On the server, we have a database dedicated to store location information such as latitude, longitude, time, speed, accuracy, uploaded time and userId.

4 Backend Processing - Real-Time / Online Clustering Algorithm

Mobile application is designed to fetch location data and we have developed backed software to process the location data. In this section, we describe clustering algorithm, which considerably processes data to give useful information.

In particular, when the user is indoors and cannot get access to GPS, there is a huge number of location data that are distributed around the actual location, as in Fig. 3. Clustering algorithm plays an important role in grouping such points to plot on the map, so as to provide better understanding as to how user uses the space.

Clustering data has been focused in many applications including data mining, statistical data analysis, computation and vector quantisation [9], pattern

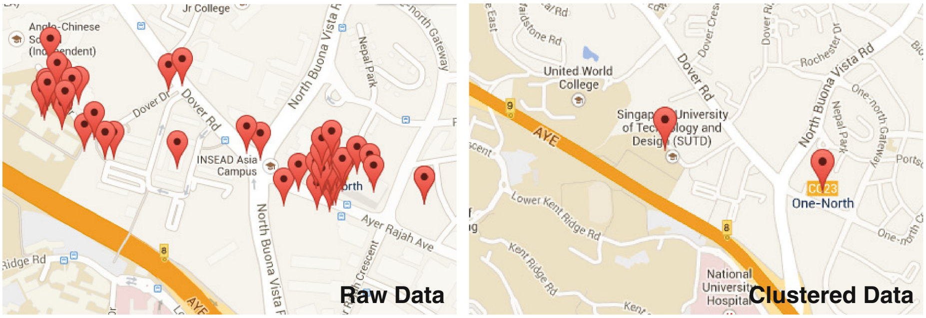

Fig. 3. Comparison of location data while indoor

```
Pseudocode
INPUT: locationDataArray          \\Data structure for storing location objects
Initialise temp to first element of locationDataArray,
for each location in locationDataArray, do:
        assign next to locationDataArray[location.next]
        assign distance to distance(temp, next)
        if distance <= sum of accuracies then:
                equate temp to average (temp, next)
                add next to tempArray
        else
                add average(tempArray) to resultCluster
                equate temp to next

end for
for each cluster in resultCluster do:
        if duration > threshold then:
                add cluster to resultNode
end for
OUTPUT: resultNode
```

Fig. 4. Pseudocode

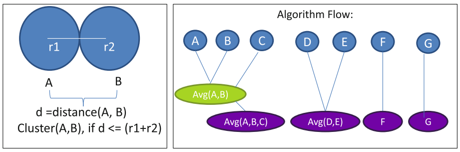

Fig. 5. Cluster algorithm flow

recognition [10] etc. K-means algorithm has had a lot of attention to combat data clustering problems from as early as 1965, when Forgy first published a standard k-means algorithm [11] to cluster data points by calculating distance between data points to predict the likelihood of data point belonging a cluster center. However, the major drawback of his approach was to predefine the value k, which is NP-Hard for arbitrary input [12]. On the basis of k-means algorithm, we have designed a clustering algorithm for non-specific number of cluster centers for arbitrary sequential input.

To enable real-time data visualisation of the places tourist visited, our algorithm is designed to have an online approach. Location object returned from android location API, contains a field for specifying accuracy of the location data returned, which states, accuracy for a location object as the radius of 68 percent confidence. In other words, if you draw a circle cantered at this location's latitude and longitude, and with a radius equal to the accuracy, then there is a 68 percent probability that the true location is inside the circle [8]. Carefully considering this, our clustering algorithm clusters location if the distance between the first location object from the second location object lies within the sum of accuracies for each location object. In other words, if the first location with accuracy radius r1 is at a distance d from the second location with accuracy radius r2, they can form a cluster if d <= r1+r2. Figure 4 describes pseudo code for our clustering algorithm.

Figure 5 describes the clustering algorithm, where the first part shows the requirement for forming a cluster and the second part shows algorithm flow for locations A, B, C, D, E, F, and G. As our algorithm is based on online clustering, we sequentially process our input. As a first step, we check if A, and B satisfy our clustering requirement, if they match, we have clustered them together. Now we check if C matches the clustering requirement with the temporary cluster formed from A and B, and since they match, we cluster A, B and C together. D does not match the clustering requirement with the cluster formed from A, B and C and hence, it will start a new cluster. E joins D to form the cluster but, F, and G, form their own clusters. After forming clusters, we highlight the location as a place for visiting if the duration is greater than the threshold. In our experiments, we vary our threshold from 5 min - 20 min.

5 Experimental Results

We conducted experiment by installing the mobile application on a Samsung Galaxy Note 10.1 tablet of a tourist to Melbourne. Smart Tourist system is able to locate all the places the tourist traveled to as well as demarcate the places according to the time they spent in each location.

Figure 6 shows a portion of Great Ocean Road with different threshold durations. We have the flexibility to adjust the threshold to be able to see different details, eg., if you set threshold to 5 mins you can see very fine details, but by

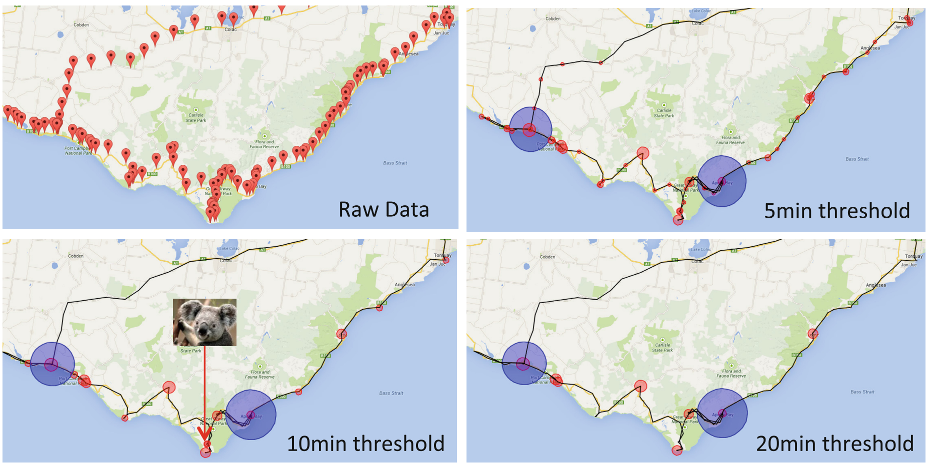

Fig. 6. Processed location data

Table 1. Places visited along great ocean road

Passive tracking		Survey	
Places	Duration (mins)	Activity	Recommendation (number of stars)
12 Rock Cafe	65	dinner	3
Lavers Hill	55	lunch	1
Twelve Apostles	40	sight-seeing	5
Lorne	40	lunch	2
Maits Rest	37	sight-seeing	4
Gibson Step	35	sight-seeing	5
Cape Otway Lighthouse	24	sight-seeing	3
Loch Ard Gorge	20	sight-seeing	5
Cape Otway	15	wild animals - koala	5
Cape Otway	15	wild animals - koala	5
Moonlight	15	sight-seeing	3
Twelve Apostles Visitor Center	10	gather information	2
Cape Otway Lighthouse Visitor Center	10	gather information	2
Bay of Island	10	sight-seeing	4
The Grotto	10	sight-seeing	4

setting it to 20 mins, you can only see those locations where tourist spent 20 mins or more.

Table 1, shows all the places visited and the time spent at each place by the tourist. First two columns are passively collected by our application, where as, third and forth are through survey, however, we can have the survey built in to the application in future.

Imagine that if over hundreds or thousands tourists install our application, we can get a very good statistics on how much other tourists have spent their time along great ocean road, the popular route they have taken, and how many nights people have stayed at that location. This information can greatly help the next tourist to plan his trip. Also, surveying the tourist that tested our application for Australia trip, he discovered that Colac is a nice town to stop by, however, as he was not aware of that and was rushing to his next spot, he missed seeing Colac. Hence, if he had the information provided by other tourists, he could have planned his trip so that he could allocate time for Colac. In addition, if the tourist provided feedback on the location, eg., marking down the location where he saw wild animals (as in Fig. 6), it would greatly help other tourists in locating wild animals.

6 Future Work and Conclusion

We have developed a software to capture and automate the information relating to tourists' experiences and time spent at each location. Going forward, we will develop a settings page so that, when a user goes on the trip, he can give information in terms of the kind of trip he is going on, whom is he traveling with, is it a family trip with young children, backpacking with friends, or together with elderly etc. By collecting information about the kind of trip, we allow tourist to plan his trip based on others who went on a similar setting.

With data coming from multiple tourists, we have to build statistics on the tourist reviews according to the information given by the tourist at the time of his trip about the kind of trip he is going on. Furthermore, we can build a survey, which tourists' might chose to fill at the end of the day, so as to make notes on their trip or voice their opinion about a tourist destination that can be shared with other tourists.

Since our algorithm currently is fetching location data at an interval of 5mins irrespective of where the tourist is, it would be preferable to design our data collection process such that we can focus on dynamic sampling rates with respect to activity recognition. For instance, when the tourist is resting at night, we can increase sampling time to 30mins to one hour, and when tourist is moving in a vehicle, we can reduce sampling to 1 min, so that we can get better knowledge of the path they travel in to go from one place to another. This approach may also be better energy efficient for mobile phones.

In this paper, we focus purely on location data for smart tourist and cluster them to give useful information, but in future we will work on using other sensor data as well. As shown in Fig. 2, we intend to do further study on the correlation of sensor data in mobility so as gain insights about tourist and their activities.

In this paper, we have specified a detailed study about our technology to improvise the experience of a tourist by learning from the experiences through passive tracking collected by peer tourists. We have developed a mobile application to collect various sensor data to passively get feedback from the tourists as well as providing them with a personal diary so that they can keep track of places they have visited. Furthermore, we have described our clustering algorithm to cater to spatial and temporal properties of our data to group raw location data into places of interest. Following the implementation, we have shown that our technology can be well suited to provide unbiased peer-to-peer reviews to tourists by passive data collection. With more mobility and location awareness, we are able to gain insights about users and their behaviours to engage them on a personal level. With connectivity like this, we are able to create new sources of value that companies, municipalities, individuals and more can leverage on.

References

1. Economic Impact of Travel and Tourism (2014). http://www.wttc.org/site_media/uploads/downloads/Economic_Impact_Summary_2014_2ppA4_FINAL.pdf

2. Smartphone Use by Tourism and Travel Consumers. http://www.targetinginnovation.com/tlx/assets/documents/uploaded/general/Final%20Draft%20Report%20Layout%20v4.pdf
3. Trip Advisor Report. http://ir.tripadvisor.com/releasedetail.cfm?releaseid=808058
4. Ali, K., Al-Yaseen, D., Ejaz, A., Javed, T., Hassanein, H.S.: CrowdITS: Crowdsourcing in Intelligent Transportation Systems. In: IEEE Wireless Communications and Networking Conference (2012)
5. Nicholas Lane, D., Mohammod, M., Lin, M., Yang, X., Lu, H., Ali, S., Doryab, A., Berke, E., Choudhury, T., Campbell, T.A.: BeWell: a smartphone application to monitor, model and promote Wellbeing. In: ACM/Springer Journal of Personal and Ubiquitous Computing (PUC), Special Issue of Cross-Community Mining (2013)
6. Guo, W., Wang, S.: Mobile Crowd-Sensing Wireless Activity with Measured Interference Power. IEEE Wirel. Commun. Lett. **2**(5), 539–542 (2013)
7. Goha, H.D., Angb, P.R., Leea, C.S.: Determining services for the mobile tourist. J. Comput. Inf. Syst. **51**(1), 31–40 (2010)
8. Android Location API. http://developer.android.com/reference/android/location/LocationManager.html
9. Wagstaff, K., Cardie, C., Rogers, S., et al.: Constrained k-means clustering with background knowledge. In: International Conference on Machine Learning, pp. 577–584 (2001)
10. Jain, A.K.: Data clustering: 50 years beyond K-means. Pattern Recogn. Lett. **31**(8), 651–666 (2010)
11. Forgy, E.W.: Cluster analysis of multivariate data: efficiency versus interoperability of classification. Biometrics **21**, 768–769 (1965)
12. Drineas, P., Frieze, A., Kannan, R., Vempala, S., Vinay, V.: Clustering large graphs via the singular value decomposition. Mach. Learn. **56**, 9–33 (2004)