A Travel Planning System Based on Travel Trajectories Extracted from a Large Number of Geotagged Photos on the Web

Kohya Okuyama and Keiji Yanai

The University of Electro-Communications, Tokyo, Japan

Abstract. Due to the recent wide spread of camera devices with GPS, the number of geotagged photos on the Web is increasing rapidly. Some image retrieval systems and travel recommendation systems which make use of geotagged images on the Web have been proposed so far. While most of them handle a large number of geotagged images as a set of location points, in this paper we handle them as sequences of location points. We propose a travel route recommendation system which utilizes actual travel paths extracted from a large number of photos uploaded by many people on the Web.

1 Introduction

Due to the recent spread of devices having cameras and GPSs such as iPhone, Android phones and some GPS-equipped digital cameras, we can easily record location information as well as digital photos. In general, photos with location information are called "geotagged photos". At the same time, some photo sharing Web sites which can handle geotagged photos such as Flickr[1] and Panoramio[2] have become popular, and the number of geotagged photos on these sites has been increasing rapidly. Since geotagged photos on the photo sharing sites can be gathered via Web API easily, recently, many researches on geotagged photos are being carried out in the field of multimedia and computer vision.

In this paper, we propose a travel planning system which utilizes a large number of geotagged photos on the Web and travel paths by many people extracted from them. In general, the places where many photos are taken by many people means tourist places drawing attention of many tourists such as historical architectures, monuments, and beautiful scenic places. By gathering many geotagged photos from the Web and analyzing them, we can get to know such places easily. In fact, many works to extract tourist places automatically from geotagged photos on the Web have been proposed so far [4, 8, 7, 10].

In addition, if a person travels through several tourist places continuously within a day or over several days, took geotagged photos at each of all the visited places, and upload them to the photo sharing sites such as Flickr or Panoramio, we can extract travel traces from a sequence of geotagged photos taken by the person. Using Web API provided by photo sharing sites, we can obtain user IDs as well as photo IDs and geotag information consisting of a set of values of latitude and longitude as metadata of photos. By obtaining meta data regarding a set of geotagged photos associated with a certain user ID, we can obtain a sequence of geotag locations which expresses a travel path of the user. This enables us to handle geotagged photos as not only a set of tourist places but also a set of travel paths.

[1] http://www.flickr.com/

[2] http://www.panoramio.com/

Then, in this paper, we extract popular tourist places and travel paths of many Web users by analyzing a large number of geotagged photos, and propose a travel route recommendation system using the extracted travel information. Our proposed system can gather, aggregate and summarize travel route information, and recommends travel routes that many Web user's preferences reflect.

This paper is organized as follows: We describe related work in Section 2. In Section 3, we explain the overview of the proposed system and its detail. In Section 4, we show experiments and examples of recommended travel paths. In Section 5, we conclude this paper.

2 Related Work

Since there are so many geotagged photos on the Web nowadays, several researches have considered the problem of selecting representative or canonical photographs for popular locations for tourists. Jaffe et al. [4] selected a summary set of photos from a large collection of geotagged photographs based on keyword tags and geotags. Simon et al. [8] have proposed a method to select canonical views for the landmarks by clustering images based on the visual similarity between two views. Kennedy et al. [5] attempted to generate representative views for the world's landmarks based on the clustering and on the generated link structure. Zheng et al. [10] built a large-scale landmark image database including 5314 landmarks using about one million geotagged photos and twenty million Web images.

Y. Zheng et al. [9] analyzed GPS log data recorded by handy GPS devices, and estimated popular human travel routes and places. Since their work used GPS log data, they can analyze precise human traces. However, the problem is that such data is very expensive to obtain, and large-scale analysis on GPS trace data is impossible, since carrying handy GPS during travel is not very common. Although travel paths extracted from geotagged photos are much coarser than GPS traces, we can obtain them much more than GPS traces instead.

As works on travel planning using geotagged photos on the Web, Cao et al. [2] proposed a travel planning system which recommends travel places based on user's preferences. This system recommends not routes and just only places.

On the other hand, X. Lu [6] proposed using travel traces extracted from geotagged photos on the Web for travel route recommendation. To generate travel paths, they used all the places extracted from geotagged photos. On the other hand, in our system, we use only geo-location information on popular tourist places and the order of traveling among them, which are represented as "trip models" in this paper. Arase et al. [1] also proposed a travel recommendation system which utilizes travel routes extracted from geotagged photos. While their objective is to recommend travel plans for the users who have not decided even areas to visit, the objective of our system is to recommend travel routes within a given area for the uses who have decided the area to visit but have not decided the tourist spots to visit within the area.

3 Proposed System

3.1 Overview

In this paper, we propose a travel route planning system, which recommends several efficient travel routes so as to visit user's favorite places. We assume that users use this system before starting trips after deciding which area they are going to travel to. Here, an "area" means a regional range over which tourist places we usually visit during one trip are distributed. For example, New York, Paris, London, Kyoto and Tokyo are examples of "areas" in this paper. Our system helps us to decide the places to visit and the order of the places to visit within the given area. How to use the system is as follows:

1. The system presents popular places in the given area with representative tags and photos, and then the user selects some favorite places.
2. The system presents several recommended routes which travel through the given places, and the user selects a favorite route.
3. The system shows the selected route on the online map.

To realize this system, as offline processing, we extract representative tourist places and common travel paths within the given area from a large number of geotagged photos collected from Web photo sharing sites in advance. This offline processing consists of the following four steps: (1) data collection (2) extraction of common tourist places (3) selection of representative photos and textural tags for each tourist place and (4) extraction and modeling of travel routes,

In this section, we explain the offline processing first, and the online processing of the proposed system next.

3.2 Data Collection and Selection

As Web photo sharing sites, we use Flickr [3] since Flickr has more than one billion geotagged photos which can be searched and obtained easily via Flickr API [4].

First of all, we obtain metadata of geotagged photos using the "flickr.photos.search" method of Flickr API. We obtain the following metadata items for each geotagged photo:

Metadata used in the proposed system

id unique ID of the photo
owner user ID who uploaded the photo
date-taken date and time of taking the photo
tags text tags for the photo
latitude, longitude geotag information
accuracy accuracy of the given geotag. Higher value means high accuracy.

Before downloading photos, we remove noise photos and select only geotagged photos suitable for extracting travel trajectories. First, we remove geotagged photos the geotag accuracy of which are less than 11. Next, to extract travel trajectories effectively, we select only the photos uploaded by the users who uploaded more than two photos taken in the same day.

Moreover, geotags of some photos are attached not by GPS but by clicking online maps, and some of them sometimes have exactly the same geotags. Such photos are also not appropriate for our work. Thus, we exclude all the pairs of the photos whose geotag locations are exactly identical but whose taken time are different by more than five minutes. As an example, we show the number of photos before and after noise removal for "user A" and "user B" in Table 1. In

[3] http://flickr.com/

[4] http://www.flickr.com/services/api/

this example, user A is estimated to be inaccurate regarding location information, while user B is estimated to be highly accurate, and probably used GPS to attach geotags to his/her photos. For this work, user A is much more useful than user B.

Table 1. The number of photos before and after noise removal

	number of original images	number of images after cleaning
user A	25189	192
user B	4793	4766

3.3 Tourist Place Detection

To detect tourist places from geotagged photos, we apply clustering for geotags of the collected photos. As clustering methods, you use a hierarchical clustering as described as follows:

1. Initially, all the geotag locations are regarded as being cluster centers.
2. Aggregate two clusters the distance between which is the closest into one new cluster. The location of the new cluster is defined as being the average location of two points.
3. If the closest distance between any two clusters become less than the predefined threshold, clustering will be finished. Otherwise, repeat from Step 2.

To compute distance D between two locations, we use the spherical distance as computed in the following equation:

$$\rho = R\cos^{-1}\{\sin\delta_A \sin\delta_B + \cos\delta_A \cos\delta_B \cos(\lambda_A - \lambda_B)\} \quad (1)$$

where R represents the radius of the earth, and $\delta_A, \lambda_A, \delta_B, \lambda_B$ represent latitude and longitude of place A and place B, respectively.

In Figure 1, we show two geotag clustering results in case that the thresholds are set as 100 meters and 400 meters. The left of the figure shows about 400 clusters in case of the 100-meter threshold , and the right shows about 100 clusters in case of the 400-meter threshold.

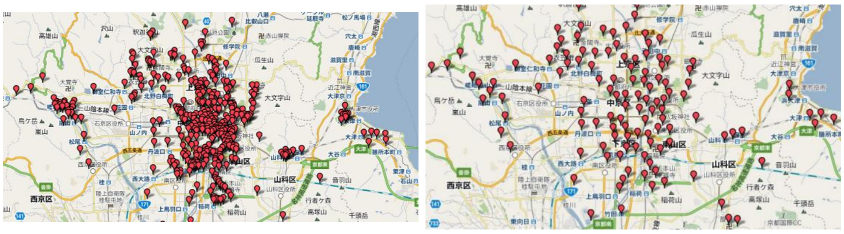


Fig. 1. (Left) Place clusters for the 100-meter threshold. (Right) Place clusters for the 400-meter threshold.

3.4 Extraction of Representative Tags and Photos

In this subsection, we describe a method to assign representative text tags and photos to each of the common tourist clusters. Representative tags and photos are helpful to explain the places both semantically and visually.

In general, text tags attached to Web photos are very various and diverse. For example, the photos of the Kinkaku Temple typically have "Kinkakuji, Kyoto, Japan, temple" as tags. Among these tags, "Japan" and " Kyoto" are not appropriate to explain the place since they means broad area. "Kinkakuji" is the best tag to explain the place of "Kinkaku temple", since it is a unique tag for the place.

To select unique tags for the places, we compute a evaluation score of uniqueness of tags $score_{c,t}$ to the places in the following equation:

$$score_{c,t} = \frac{N_c(t)}{\sum_{c \in C} N_c(t)}$$

where C, c, t, and $N_c(T)$ represents all the clusters, a cluster, a tag, and the number of photos having tag t in cluster c, respectively. If $score_{c,t}$ is more than the pre-defined threshold, the tag t is regarded as being one of representative tags of the cluster c.

In Table 2 and Table 3, we show two examples before and after the tag selection on the clusters including the Kinkaku temple and the Kiyomizu temple. In these examples, some tags which means broad areas such as "japan" and "kansai" are eliminated successfully.

Table 2. Frequent tags before the selection, and selected tags on the cluster including the Kinkaku temple.

before selection	kinkakuji japanesebuses japan kansai kyoto ofriceandzen
after selection	kinkakuji japanesebuses

Table 3. Frequent tags before the selection, and selected tags on the cluster including the Kiyomizu temple.

before selection	october japan october kyoto kiyomizudera 2009 kiyomizutemple
after selection	kiyomizudera kiyomizutemple

Next, we select some representative photos for each cluster by employing local-feature-based image analysis. We decide representative photos based on the number of matched local feature points. As local features, we use SURF (Speeded-Up Robust Features) [3]. We extract SURF descriptors from each of the photos, and search for matched local point pairs within the same cluster. We select the top five photos as representative ones in terms of the number of matched points in each cluster. Two figures in Figure 2 show local feature matching between two photos of the Kinkaku temple taken from the different angles and between two photos of the Kinkaku temple and the Kiyomizu temple. In general, the pair of the photos of the same landmark bring much more matched points than the pair of the different landmarks. As a result, the landmark taken in many photos gathers many matched points and it is selected as representative one.

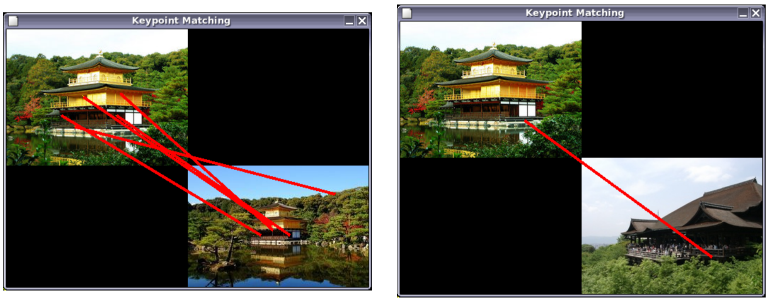


Fig. 2. SURF point matching. (Left) between two photos of the Kinaku Temple. (Right) between the Kinkaku and the Kiyomizu temple.

3.5 Modeling Travel Trajectories

In the proposed system, after selecting tourist place clusters and their tags and photos, we extract travel trajectories of many people from geotagged photos , and generate “trip models” from them which represent canonical move sequences among tourist place clusters.

At first, we extract geotag sequences from geotagged photos taken by one user within a certain day in the time order, and gather them regarding many users and many days as a set of one-day travel trajectories. We call an one-day travel trajectory as “a trip”. We show the places where a certain user took geotagged photos within the same day in the left of Figure 3, and their travel path in the right of Figure 3.

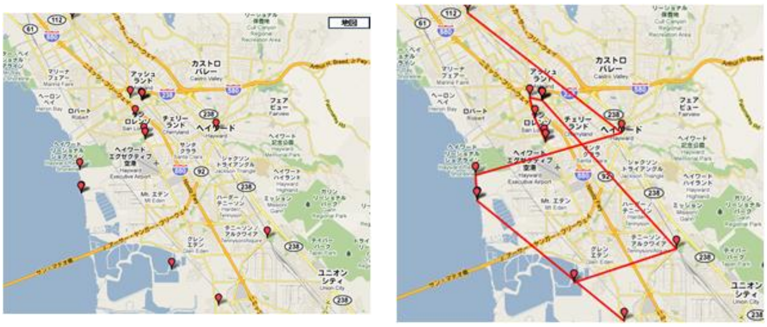


Fig. 3. (Left) Photo places of a certain user in a day. (Right) Travel sequences.

Some “trips” include many geotag places densely, while some “trips” include some places sparsely most of which are popular tourist places. In addition, some ”trips” might include many geotag locations around the same landmarks. In this way, the resolution of geotags varies depending on users greatly. Then, to make the resolution of “trips” even and to remove redundant places, we convert a “trip” trajectory, which is a sequence of geotagged places, into a “trip model”, which is a sequence of the moves between common tourist places detected in

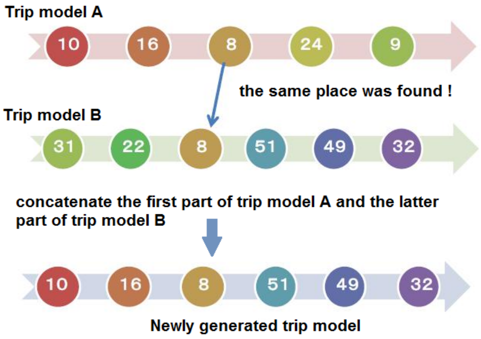

Trip model A
10
16
8
24
9
the same place was found !
Trip model B
31
22
8
51
49
32
concatenate the first part of trip model A and the latter part of trip model B
10
16
8
51
49
32
Newly generated trip model

Fig. 4. An example of generating a new "trip model".

the previous section. In this paper, all the travel paths are represented as "trip models".

Although we can extract "trip models" from geotagged photos in this way, the number of "trip models" is not sometimes enough. Then, in the next step, we generate new "trip models" by combining several "trip models". The proposed method to generete new "trip models" shown in Figure 4 is as follows:

1. Search for the two "trip models" both of which include the same tourist places.
2. Concatenate the first part of one trip model before the place and the latter part of the other trip model after the place.
3. If the generated "trip model" does not include duplicate places, the "trip model" is regarded as being valid. Otherwise, it will be discarded.

Note that to prevent loops from being made in trip models, we limit the same place to being included once in one "trip model".

In Figure 4, a new "trip model" is generated by linking the first part of "trip model A" with the latter part of "trip model B". In addition, we can concatenate the first part of "trip model B" and the latter part of "trip model A".

3.6 Online System

In this subsection, we explain online processing on a route recommendation system which makes use of common tourist places and "trip models" extracted from geotagged photos. The online processing of the system consists of the following three steps:

1. Selection of tourist places where a user like to visit
2. Presenting travel route candidates and selection from them
3. Presenting the selected travel route on the map

Each of three figures in Figure 5 corresponds to each of the above three steps.

Selection of places In the first step, the system shows common tourist places with their tags and representative photos, and asks a user to select some places where he/she like to visit. As help for a user to select places, the system shows

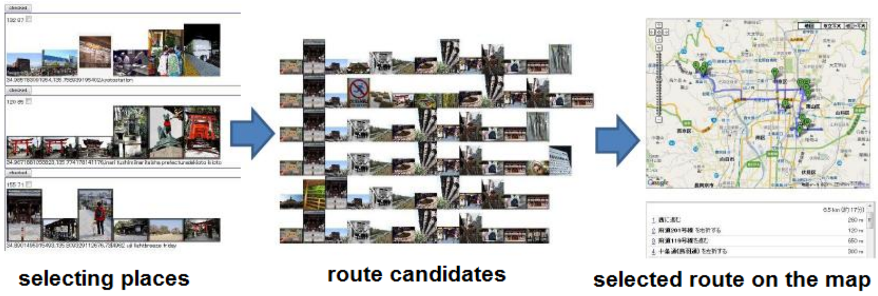


Fig. 5. Three steps in the online system

some information on each place which are obtained via Yahoo! Local Search API API [5] as well.

Showing and selection of "trip models" The system searches the "trip model" database for the routes including as many tourist places the user selected as possible. To search for "trip models" quickly, we prepare an search index on "trip models" regarding each of common tourist places in advance.

Presenting a travel route on the map In the last step, the system shows the selected "trip model" on the map. "Trip models" do not contain road information between tourist places, but contain only information on sequences of tourist places. Then, we obtain common road routes between the tourist places by using the Directions Service of Google Maps API [6]. Using this service as well as basic function of Google Maps API, we can present the selected "trip model" on the map as shown in the right figure of Figure 5.

4 Experiments

To gather a large number of geotagged images within areas where people travel around in one day, popular tourist areas are appropriate. As a target area in the experiment, we selected "Kyoto" which is one of the most popular tourist areas in Japan.

We gathered twenty thousand geotagged photos taken in the Kyoto area from Flickr via Flickr API. After noise removal described in Section 3.2, we collected 1805 geotagged photos uploaded by 162 unique users. After carrying out hierarchical clustering with the 400-meter threshold in terms of the radius of clusters, we obtained 154 tourist place clusters and 18,742 "trip models", part of which are shown in Table 6.

Figure 7 shows a part of common tourist place candidates with representative tags and photos, and address information on the locations obtained from Yahoo! Local Search API. The tourist places are shown in the descending order of the number of the cluster members of geotagged photos, that is, in the order of popularity.

As a case study, we selected four places, "Kyoto Station", "Fushimi temple", "Uji bridge" and "Nijo castle". As route candidates that go through the given

[5] http://developer.yahoo.co.jp/webapi/map/localsearch/v1/localsearch.html

[6] http://code.google.com/intl/ja/apis/maps/

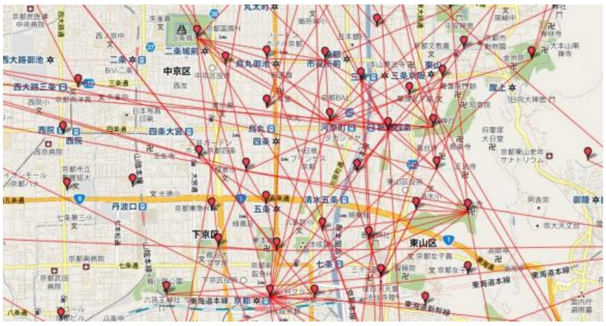


Fig. 6. Extracted common tourist places and “trip models” in “Kyoto”.

Table 4. The number of visiting tourist places, and the total and average distances of the three routes.

Route candidate	num. of places	Total distance (km)	Avg. dist. between places (km)
Trip 1	9	29.8	3.73
Trip 2	15	52.7	3.76
Trip 3	11	54.1	5.41

four places, three route, “Trip 1”, “Trip 2” and “Trip 3”, are presented in the ascending order of the total moving distance. When clicking each of route candidates, each of the three routes are displayed on the map as shown in Figure 8, Figure 9 and Figure 10, respectively. Table 4 shows the number of common tourist places included in the route candidates, the total moving distances and the average moving distances between the common tourist places.

From Table 4, a user who wants to move quickly between selected places will select “Trip 1” or “Trip 2”, while a user who wants to visit as many tourist places as possible might select “Trip 3”. In this way, selection from the route candidates depends on the user’s preference greatly. Thus, the important thing is presenting as many route candidates and their additional information as possible which suit user’s preference conditions.

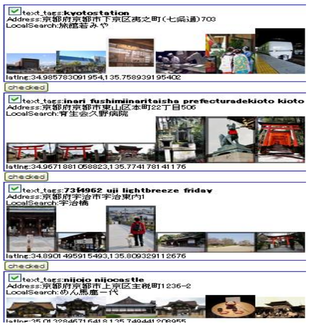
5 Conclusions

In this paper, we proposed a travel route recommendation system based on sequences of geotagged photos on the Web, which presents several travel route candidates with representative tags and photos by selecting tourist places where a user like to visit. To gather information on travel routes, we proposed “trip models” represented by the order sequences of tourist places. In the experiments, we build a “trip model” database on the Kyoto area as a case study.

As future work, we plan to apply many other areas over the world than Kyoto and extend the system to recommend travel routes taking account of travel time.

References

1. Arase, Y., Xie, X., Hara, T., Nishio, S.: Mining people’s trips from large scale geo-tagged photos. In: Proc. of ACM International Conference Multimedia (2010)
2. Cao, L., Luo, J., Gallagher, A., Jin, X., Han, J., Huang, T.: A worldwide tourism recommendation system based on geotagged web photos. In: Proc. of IEEE International Conference on Acoustics, Spech and Signal Processing (2010)



☑ text_tags:**kyotostation**
Address:京都府京都市下京区夷之町(七条通)703
LocalSearch:旅館若みや
latlng:34.985783091954,135.758939195402
checked

☑ text_tags:**inari fushimiinaritaisha prefecturadekioto kioto**
Address:京都府京都市東山区本町22丁目506
LocalSearch:育生会久野病院
latlng:34.9671881058823,135.774178141176
checked

☑ text_tags:**73f4962 uji lightbreeze friday**
Address:京都府宇治市宇治東内1
LocalSearch:宇治橋
latlng:34.8901495915493,135.809329112676
checked

☑ text_tags:**nijojo nijocastle**
Address:京都府京都市上京区主税町1236-2
LocalSearch:めん馬鹿一代
latlng:35.0132846716418,135.749441208955

Fig. 7. Part of place candidates.

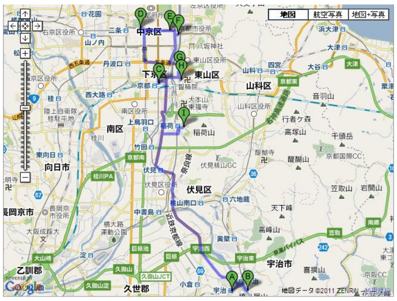


Fig. 8. The route of "Trip1". B: Uji bridge. C: Kyoto St. D: Nijo castle. I:Fushimi temple.

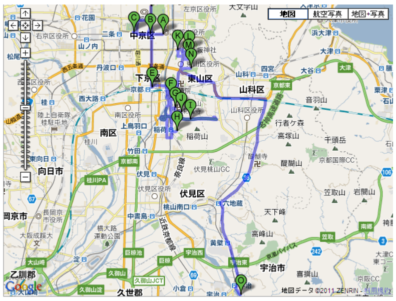


Fig. 9. The route of "Trip2". C: Nijo castle. E: Kyoto St. H: Fushimi temple. O: Uji bridge.

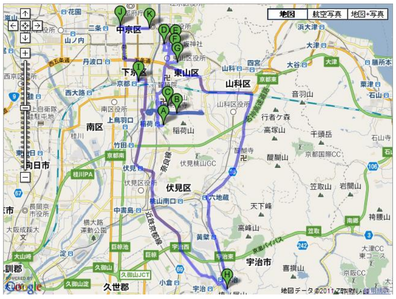


Fig. 10. The route of "Trip3". A: Fushimi temple. H: Uji bridge. I: Kyoto St. J: Nijo castle.

3. Herbert, B., Andreas, E., Tinne, T., Luc, G.: Surf: Speeded up robust features. Computer Vision and Image Understanding pp. 346–359 (2008)
4. Jaffe, A., Naaman, M., Tassa, T., Davis, M.: Generating summaries and visualization for large collections of geo-referenced photographs. In: Proc. of ACM SIGMM International Workshop on Multimedia Information Retrieval. pp. 89–98 (2006)
5. Kennedy, L., Naaman, M.: Generating diverse and representative image search results for landmarks. In: Proc. of the ACM International World Wide Web Conference. pp. 297–306 (2008)
6. Lu, X., Wang, C., Yang, J., Pang, Y., Zhang, L.: Photo2Trip: Generating travel routes from geo-tagged photos for trip planning. In: Proc. of ACM International Conference Multimedia (2010)
7. Quack, T., Leibe, B., Gool, L.V.: World-scale mining of objects and events from community photo collections. In: Proc. of ACM International Conference on Image and Video Retrieval. pp. 47–56 (2008)
8. Simon, I., Snavely, N., Seitz, S.M.: Scene summarization for online image collections. In: Proc. of IEEE International Conference on Computer Vision (2007)
9. Zheng, Y., Zhang, L., Xie, X., Ma, W.Y.: Mining interesting locations and travel sequences from gps trajectories. In: Proc. of ACM World Wide Web Conference (2009)
10. Zheng, Y., Zhao, M., Song, Y., Adam, H., Buddemeier, U., Bissacco, A., Brucher, F., Chua, T., Neven, H.: Tour the world: building a web-scale landmark recognition engine. In: Proc. of IEEE Computer Vision and Pattern Recognition (2009)