Trip Mining and Recommendation from Geo-tagged Photos

Huagang Yin[1]*, Changhu Wang[2], Nenghai Yu[1], Lei Zhang[2]
[1]MOE-MS Key Lab of MCC, University of Science and Technology of China, Hefei, China
[2]Microsoft Research Asia, Beijing, China
yinhg@mail.ustc.edu.cn, {chw, leizhang}@microsoft.com, ynh@ustc.edu.cn

***Abstract*—Trip planning is generally a very time-consuming task due to the complex trip requirements and the lack of convenient tools/systems to assist the planning. In this paper, we propose a travel path search system based on geo-tagged photos to facilitate tourists' trip planning, not only for where to visit but also how to visit. The large scale geo-tagged photos that are publically available on the web make this system possible, as geo-tagged photos encode rich travel-related metadata and can be used to mine travel paths from previous tourists. In this work, about 20 million geo-tagged photos were crawled from Panoramio.com. Then a substantial number of travel paths are minded from the crawled geo-tagged photos. After that, a search system is built to index and search the paths, and the Sparse Chamfer Distance is proposed to measure the similarity of two paths. The search system supports various types of queries, including (1) a destination name; (2) a user-specified region on the map; (3) some user-preferred locations. Based on the search system, users can interact with the system by specifying a region or several interest points on the map to find paths. Extensive experiments show the effectiveness of the proposed framework.**

Keywords- Geo-tagged Photos, Path Mining, Search System, Trip Planning.

I. INTRODUCTION

The prosperity of tourism has made travel increasingly popular in people's everyday life. When a tourist wants to start a new trip, it is very natural to make plans prior to the trip. But travel decision making is a very complex task, including many factors, such as travel destination, travel duration, visiting time, travel budget, destination preference, season, weather, etc. It is not trivial to build a model/system to incorporate all these factors. Or, it is even hard for users to express their travel intention.

Although a user can search for related travel guides or ask questions in web-based communities, it is very time-consuming to manually summarize tens of travelogues and find a good travel plan. Since the information provided by other people's travelogues is usually unstructured and varies from person to person, from language to language, it is extremely hard for users to follow.

When a user wants to have a new trip, he/she usually has two questions:

(1)**What is the travel sequence**: Travel sequence means the travel order of attractions. For example, when a user chooses Xi'an, which is the capital of thirteen China dynasties, as the destination, he/she needs to decide the travel sequence by choosing from about 192 attractions.

(2)**What is the travel path**: Travel path means the route within an attraction. Before tourists enter an attraction, it is very important to choose a suitable path in order to visit the most beautiful sceneries, especially when their time is limited.

With the development of Web 2.0 technologies, more and more people have willingness to share their travel photos on the Web. These publically available photos on the web encode rich travel-related information and are sufficient to cover most of the countries in the word. These photos not only show the beautiful scenes of other tourists have seen, but also encode where and when the scenes were captured. From these photos we can get tourists' travel paths recorded by geo-tagged photos. A travel path composed by photos encodes geographic information, time information and provides the preview of scenery along the path.

Geo-tagged photos are used to support applications such as landmark discovery/recognition [7] and location explorer [8]. However, they only consider the geo-information of photos themselves. [2] proposed a framework named as Photo2Trip and tried to automatically generate travel routes by leveraging the time sequence of geo-tagged photos. But travel sequences generated by dynamic programming are not real sequences which cannot meet users' complex travel requirements. Meanwhile, there might be too many travel paths within one attraction, making it not easy for users to find a proper path to meet their preference.

In fact, the aforementioned two questions are two geography levels of how to visit. The travel sequence is higher and the travel path lower. They can be modeled in a unified framework. In this paper, we propose a framework based on geo-tagged photos to help users make travel decisions. After collecting paths of the two geography levels, we setup a path search system to manage the paths and develop an application to address these two questions.

As shown in Figure 1, our framework contains the following three modules:

- Path Collection: With these geo-tagged photos we can mine where users have visited, when they visited, and the time they have spent.

*This work was performed at Microsoft Research Asia.

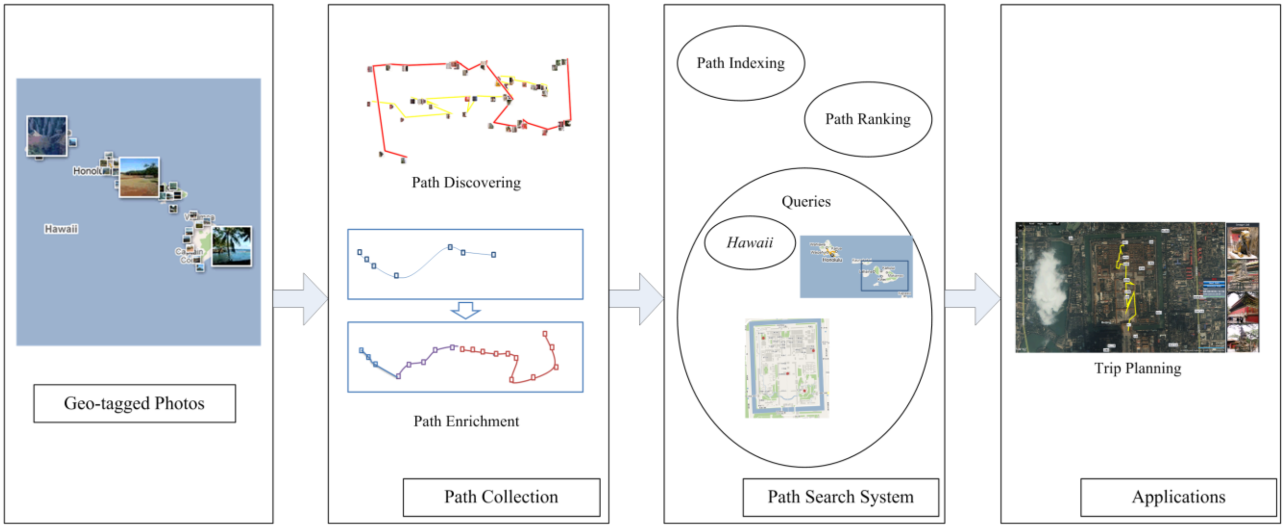

Figure 1: The overview of the proposed framework. Travel paths were mined from about 20 million geo-tagged photos. Then a search system was built to manage the paths and a useful application was implemented.

- Path Search System: We present a path search system to manage the mined paths.
- Applications: Based on the system, applications could be developed to help users plan trips.

II. PATH COLLECTION

In this section, we present how to mine paths from geo-tagged photos. Because the path is discrete and incomplete, algorithms are developed to enrich the paths.

A. Path Discovery

Useful information can be inferred from the geo-tagged photos. The most important information is the travel path of users, from which we can know not only where and when they visited but also how they visited.

As shown in Figure 2, photos with geo-tags and time stamps present where and when the photos were taken. An algorithm named Path Discovering Algorithm was proposed to discover paths from geo-tagged photos. Photos were grouped by their owner and ordered by their taken time. If we place one's photos on the map and link them one by one, we can get a travel path of the user.

Because tourists do not take photos continuously and only share the subset of their photos, paths mined from photos cannot reveal the true path of tourist. For each path, we hope to measure its quality. A function is defined to measure how "ideal" a path is. An "ideal" path is defined as a path along which a virtual user takes photos all the time and at all positions after he/she enters a destination. That means an ideal path should have an unlimited photo density and a relatively large span. We use the following equation to calculate the path quality [2]:

$$Q_P = 1 - exp\{-L_P \times \rho_P\}, \qquad (1)$$

where L_P denotes the length of the path, ρ_P denotes the density of the path. A longer path with a larger density has a bigger quality score.

From the travel paths, other useful attributes of the path can be mined, which are illustrated in Table 1. The time related attributes of a path help us plan the travel time. The path occurrence time is important because from other tourists' experience we can know whether an attraction should be visited in summer or winter. From other tourists' duration in an attraction we can estimate how long we will spend in the attraction.

TABLE I. ATTRIBUTES OF PATH P

X_P	Path location	Center of the path, roughly shows where the path is.
T_P	Path occurrence time	Taken time of the path's first photo, shows when the tourist had the journey.
D_P	Path time duration	The time from the path's first photo to the last,
Q_P	Path quality	Score calculated by Equation (1), shows how ideal (attractive) the path is.
P_P	Path photos	Photos which compose the path.

B. Path Enrichment

Because tourists usually only share a subset of their photos, paths mined from photos cannot repeat the true path of tourist. But by combining different users' paths, we can get better results. Maybe one user shares the upper part of an ideal path and another user shares the lower part. Maybe one user shares a long path whose photo density is far from being satisfactory and another user shares part of the long path which has a satisfy photo density.

Based on the path search system introduced in the next section, we introduce the Search Based Path Enrichment (SBPE) algorithm to improve both the length and the density of paths. As shown in Figure 4, the big jump of a path or the end of a path are considered as queries and used to search candidate fragments. Then the original paths are enriched by replacing the queries with the candidate fragments.

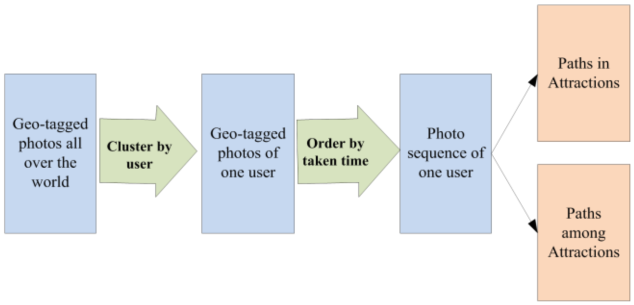

Figure 2: The path discovery procedure.

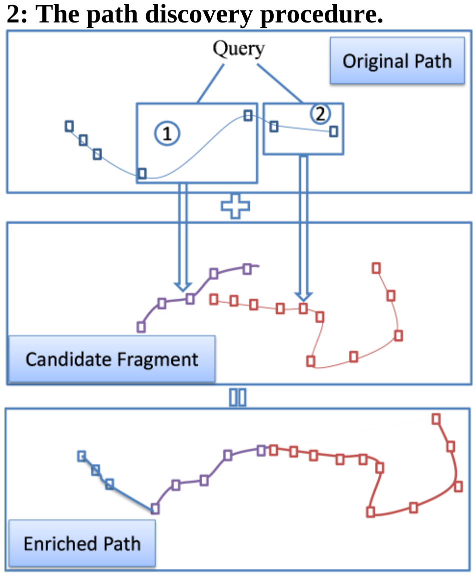

Figure 3: Search-based Path Enrichment algorithm. The big jump (query 1) and the end (query 2) of the original path are considered as queries and used to search candidate fragments. Then the original paths are enriched by replacing the queries with the candidate fragments.

III. PATH SEARCH SYSTEM

In this section, we present a path search system to manage the mined paths.

A. *Path Indexing*

From Table I we can see that a path has the following attributes: path location, path occurrence time, path duration, etc. As trip planning is location-based service, the most important attribute of a path is its location, e.g. latitude and longitude. So spatial index is used to manage the path about where the path is on the earth. We use the center of a path to represent its location on the earth.

The quadtree is used to index the paths. A quadtree is a tree in which each internal node has exactly four children. With the geographic coordinate system, every location on the earth can be specified by a set of numbers. A pair of latitude and longitude defines a point in a two-dimensional plane.

Because the density of paths on the earth varies differently, many of the leaves of a complete quadtree would be empty. As the size of the bounding box represents an area on the earth, we do not need to subdivide squares if they contain only one particle. An adaptive quadtree algorithm is used to index the paths (Figure 4).

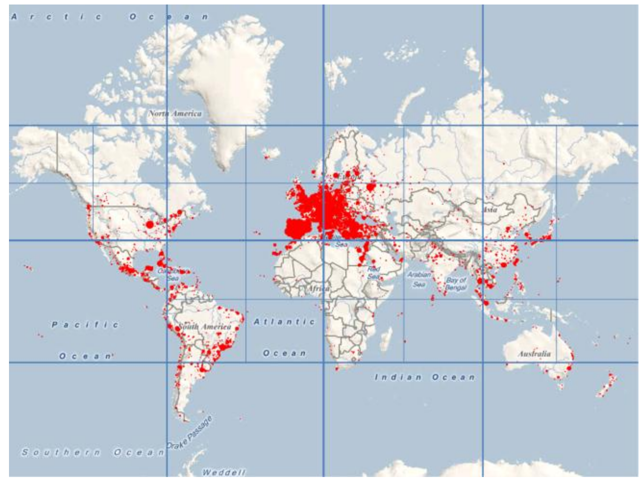

Figure 4: An adaptive quadtree to index the paths in attractions. The red circles show the distribution of paths on the earth.

B. *Path Similarity*

It is very important to compute the similarity in a search system. The chamfer distance [10] is usually used to measure the distance of contours. However, the paths collected in the previous section are discrete, so we propose the Sparse Chamfer Matching and the Directed Sparse Chamfer Matching to measure the similarity of paths.

1) Chamfer Distance

The chamfer distance function was proposed to measure the similarity of two contours [10]. If paths are not discrete, we can use chamfer distance to measure the distance of two paths.

Two paths can be considered as two sets of points, the asymmetric distance from path P_i to P_j is calculated by the following formula:

$$d_{cham}(P_i, P_j) = \frac{1}{|P_i|} \sum_{X_{P_i} \in P_i} \min_{X_{P_j} \in P_j} \left\| X_{P_i} - X_{P_j} \right\|_2, \qquad (2)$$

where $|P_i|$ denotes the number of points in path P_i, and $\|\cdot\|_2$ the L_2-norm. The chamfer distance measures the mean distance of points in P_i to their closest points in P_j. The distance can be efficiently computed via distance transform [10]. The distance transform of each path can be pre-computed offline. The distance transform of path P is as follows:

$$DT_P(X) = \min_{X_P \in P} \|X - X_P\|_2\,, \qquad (3)$$

and,

$$ADT_P(X) = arg \min_{X_P \in P} \|X - X_P\|_2 \qquad (4)$$

selects the closest point in P.

Equation (3) and (4) can be computed in O(n) time [11]. For each path, the distance transform only needs to be calculated once. When calculating the chamfer distance from P_i to P, the operation in (2) becomes:

$$d_{cham}(P_i, P) = \frac{1}{|P_i|} \sum_{X_{P_i} \in P_i} DT_P(X). \quad (5)$$

The distance transform of path P was pre-computed at first. Then, the chamfer distance from P_i to P is the average of the overlap numbers.

2) Sparse Chamfer Matching

As the path recovered from photos is sparse, the points generated by linking two photos are not exactly the locations that a tourist has visited. Thus we proposed the Sparse Chamfer Matching (SCM) algorithm based on Gaussian Mixture Model to measure the similarity of two paths. The SCM algorithm only uses the photos and ignores the lines which link the photos.

Give a photo P_P of a tourist, we know that the tourist has been to the location X_{P_P} of P_P. We use a Gaussian function to measure the probability of the tourist has been to a location X around P_P:

$$G_{P_P}(X) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\left(\left\|X - X_{P_P}\right\|_2\right)}{2\sigma^2}}. \quad (6)$$

Give a path P of a tourist and photos $P_P \in P$, the probability the tourist has been to a location is defined as

$$f_P(X) = \frac{1}{|P|} \sum_{P_P \in P} G_{P_P}(X). \quad (7)$$

The asymmetric similarity from path P_i to P_j is evaluated as the following formula:

$$sim_{SCM}(P_i, P_j) = \frac{1}{|P_i|} \sum_{X_{P_i} \in P_i} f_{P_j}(X_{P_i}). \quad (8)$$

Like the Chamfer Distance, $f_P(X)$ can be pre-computed offline. The calculation of $sim_{SCM}(P_i, P_j)$ is a look-up and average process.

C. Supported Queries

Traditional text-based search techniques are not always effective. We investigate new path-based search methods to allow a user to interactively search for travel paths. As Figure 1 illustrated, our system supports three types of queries, including: (1) a destination name; (2) a region on the map; (3) some interest points on the map.

A region is a rectangle on the map and can be represented by the range of latitude and longitude. Users can interact with a virtual map, drawing a rectangle anywhere of any size. With different geography size of the region, it can be a country, a city or an attraction.

Users may have special interests about some locations on the map. For example, if they favor some attractions in a city or some sub-attractions in an attraction, the search system can search for paths which include the specified locations.

If a user specifies his/her interest about points X_u, the Sparse Chamfer Distance is used to match paths:

$$\{P | sim_{DSCM}(X_u, P) > sim_{thr}\}, \quad (9)$$

where sim_{thr} is a threshold.

D. Path Ranking

Ranking is also an important component of the path search system. For each query, the search system will find a lot of paths. Equation (1) has defined a score to measure the quality of paths. When ranking the paths, the popularity of the paths is also very important. An algorithm using Random Walk with Restarts (RWR) is proposed to rank the candidate paths. The algorithm not only uses paths' relationship by defining a similarity, but also leverages the quality information of paths. With the RWR algorithm, we can compute a PathRank (like PageRank[9]) for each paths offline.

Each candidate path p_i is considered as a vertex of a graph G. All vertices of G are fully connected and the weight between two vertices is the similarity of the two paths. Two paths are more similar if they have a smaller distance. The weight of the edge between path p_i and path p_j can be calculated by Equation (8).

The RWR algorithm [9] models a user's random walk on a graph. Assume that there is a random walker starting from node p_i with a certain probability. At the next step, the walker has two choices. One is to randomly choose an out-link to follow. The other choice is to jump to p_j with probability $c \times v_j$, where v is the restart vector and c is the probability of restarting the random walk.

Assume that G is a graph with N vertices $p_i (i \in [1, N])$. A is the adjacency matrix of G and has been column-normalized to ensure the sum of each column is one. The quality scores of each path are considered as the restart vector v, which is also normalized such that the sum of v is one. So the path with a higher quality will be chosen with a higher probability. Our goal is to get the steady state probability of all vertices, which is denoted by u. Then the steady state probability satisfies the following formula:

$$u = (1 - c)Au + cv. \quad (10)$$

Therefore

$$u = c(I - (1 - c)A)^{-1} v. \quad (11)$$

From the definition of the adjacency matrix of G we can see if a path p_i is similar to other paths, the walker will choose it with a higher probability. This path can be seen as a popular path. So the RWR algorithm can find more popular paths which also have a higher quality.

IV. APPLICATIONS AND EXPERIMENTS

A. Data Set

Geo-tagged photos are the cornerstone of our framework. With more geo-tagged photos, more paths will be mined and indexed. Then users may get more satisfactory paths when they search in our system.

20,287,695 geo-tagged photos and related attributes were crawled from Panoramio.com. Each photo has its owner, title, tags, upload time, taken time, latitude, longitude, etc. These photos belong to 1,267,123 users. The largest number of photos one user shared is 16,407 and the smallest is 1.

As described in section II.A, individuals' true paths were mined from the crawled geo-tagged photos. At a lower level, 1,372,809 travel paths in attractions were discovered. Figure 4 illustrates the distribution of paths on the earth. At a higher level, 97,505 travel sequences among attractions were discovered.

B. Text-Based Path Recommendation

Users can input a destination name and then get the recommended paths within the query destination. Figure 5 illustrates a path recommended for query "the Forbidden City". High-quality photos taken in that destination are also shown to users. Users can preview the photos to decide whether they like this destination.

At the same time, some statistic information of the path is also provided to users. For example, the distribution of the time duration along the path can tell how long will be spend in this destination. For each text query, the number of the returned paths indicates the times users have visited this destination. We define the popularity of a destination by the total path number of the destination. With this popularity, we can recommend to users what the hottest destination(s) is/are. The popularity can be calculated in terms of seasons or months, and thus we can tell users whether March or October is the best travel time. For example, the Forbidden City has the most tourists in August. Of course users can choose an off season to visit the Forbidden City if they like.

C. Region-Based Path Recommendation

When users navigate the earth on a virtual map, they can specify any region they are interested on the map. It is very convenient to show paths in the view a user is watching. The region can be a country, a city or an attraction. Figure 5 illustrates a path for the region of the Forbidden City. The search results are given in the same way as in text-based path recommendation.

The power of this application is that the region can be anywhere on the map. For example, if a user only wants to visit places in the east half of London City, they can specify the region and then get attractions as well as travel sequences just in the east half city.

We have evaluated different ranking methods. Since there is no ground-truth about the recommendation results, we conducted a user study to evaluate the text/region-based path recommendation.

Twenty graduate students were asked to complete the following task. They were asked to input a familiar destination name and then score the top 5 paths in the search results using score 1 to 5. Four aspects were asked and evaluated:

Representative: to what extent the recommended path reflects the culture and characteristics of the destination;

Comprehensive: to what extent the recommended path offers a comprehensive view of the destination;

Novelty: to what extent the recommended path has interested you after you view the photos along the path;

Satisfaction: the overall score about the recommended path.

To evaluate the performance of our recommendation, the following algorithms were compared: (1) ranking the recommendation results by the count of photos (RBP); (2) ranking the recommendation results by the quality proposed in Equation 1 (RBQ); (3) ranking the recommendation results by the RWR algorithm proposed in Section III.D (RBRWR). For each algorithm, the original paths were evaluated first, which were denoted by RBP-O, RBQ-O, RBRWR-O, respectively; the enriched paths were evaluated later, which were denoted by RBP-E, RBQ-E, REBWR-E, respectively. Methods RBQ-O are the path ranking methods in [2].

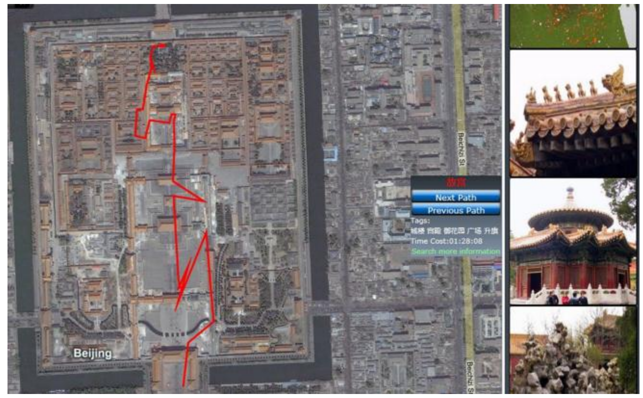

Figure 5: Path recommended for text query "the Forbidden City" or the region of the Forbidden City. In the search system, text queries will be transformed to regions.

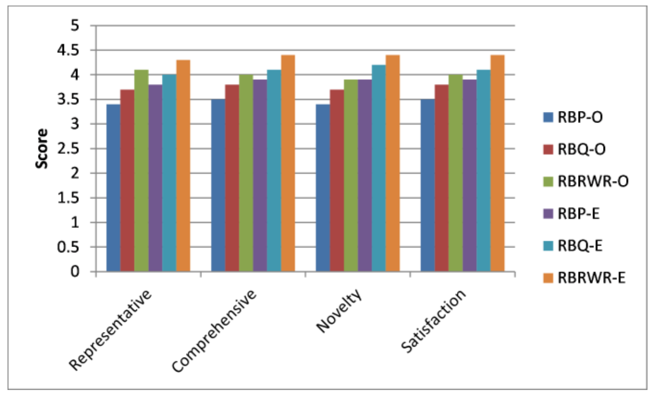

Figure 6: Comparison results of different algorithms performed on original paths and enriched paths.

The results are shown in Figure 6, from which we can see that the RWR algorithm performed on enriched paths give users the best experience. For the paths themselves, enriched paths are better than the original paths. The RWR algorithm performs best in the ranking algorithms. Experimental results show that our methods greatly outperform that of [2].

D. Interesting Place-Based Path Recommendation

Either heard from other people or attracted by some photos, users may have special interests in certain locations. In this case, these locations might be the most important facts when users plan their trip.

The locations could be submitted as a query to the search system, and popular paths which pass these locations are returned as the search results. For example, if a user selects 5 attractions out of all 192 attractions of Xi'an City (in China) as their favorite locations, typical paths which pass all the 5 attractions will be provided to tell which paths other people went along when they visited these attractions. Users can also specify only one or two locations in an attraction that they think they should not miss.

Figure 7: Path recommendation of user specified places(two red pins). The user thinks he/she should not miss the Hall of Supreme Harmony and the Palace of Tranquil Longevity in the Forbidden City.

Figure 7 illustrates a recommended path of user specified places. In the Forbidden City, a user heard that he/she should not miss the Hall of Supreme Harmony and the Palace of Tranquil Longevity. So he/she added two pins on the map and got filtered paths within the Forbidden City.

V. RELATED WORKS

Before the prosperity of geo-devices, users always searched for related travel guides through the existing search engine or travel sites. But it is time-consuming to manually summarize tens of travelogues and find a good travel plan.

GPS trajectories directly show the true path of users. [1] collected 107 users' GPS logs and mined interesting locations and travel sequences form the logs. But it is not easy to collect such trajectory data from numerous users due to their privacy concerns. Therefore, there are not enough GPS trajectories to support trip planning in a worldwide scale.

Different from [1], [2][3] used geo-tagged photos as the data source and tried to automatically suggest travel routes for users. [4] crawled 5.7 million geo-tagged photos and performed photo trip pattern mining. It focuses on detecting people's frequent trip patterns, but did not make plans.

[5] proposed a photographer behavior model to estimates the probability of a photographer visiting a landmark. They incorporated user preference and presented location information into the probabilistic behavior model by combining topic models and Markov models.

Working with more than 8 million of geo-tagged photographs, the goal of [6] is to meet the needs of a tourist who specifies a starting location together with a bounded travel distance. The Antourage system proposed in [6] generate a tour that visits the popular sites in a region.

Different from these works, we think that the task of travel decision making includes many factors and cannot be easily solved based on a model. So we build a path search system in this work to help users plan their trips. Users can search for other people's true travel paths and plan their own ones.

VI. CONCLUSIONS

In this work, we introduced a path search system to facilitate tourists' trip planning, not only where to visit but also how to visit. The paths were discovered from more than 20 million geo-tagged photos which encoded rich travel related metadata. Because the paths may be discrete and incomplete, a Search Based Path Enrichment algorithm was proposed to get better paths. Moreover, the Sparse Chamfer Distance was proposed to measure the similarity of two paths. A novel application was introduced, in which users could interact with the search system and plan their trips at home before they travel

ACKNOWLEDGMENT

Huagang Yin and Nenghai Yu were partially supported by Fundamental Research Funds for the Central Universities (No. WK2100230002), National Natural Science Foundation of China (No. 60933013), and National Science and Technology Major Project (No. 2010ZX03004-003).

REFERENCES

[1] Yu Zheng, Lizhu Zhang, Xing Xie, Wei-Ying Ma Mining Interesting Locations and Travel Sequences from GPS Trajectories. WWW 2009

[2] Xin Lu, Changhu Wang, Jiang-Ming Yang, Yanwei Pang, and Lei Zhang. Photo2Trip: Generating Travel Routes from Geo-Tagged Photos for Trip Planning. ACM Multimedia 2010

[3] An-Jung Cheng, Yan-Ying Chen, Yen-Ta Huang, Winston H. Hsu, and Hong-Yuan Mark Liao. Personalized travel recommendation by mining people attributes from community-contributed photos. ACM Multimedia 2011

[4] Yuki Arase, Xing Xie, Takahiro Hara, and Shojiro Nishio. Mining people's trips from large scale geo-tagged photos. ACM Multimedia 2010

[5] Takeshi Kurashima, Tomoharu Iwata, Go Irie, and Ko Fujimura. Travel route recommendation using geotags in photo sharing sites. CIKM 2010

[6] Saral Jain, Stephan Seufert, and Srikanta Bedathur. Antourage: mining distance-constrained trips from flickr. WWW2010

[7] Yan-Tao Zheng, Ming Zhao, Yang Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, Tat-Seng Chua, H. Neven. Tour the world: building a web-scale landmark recognition engine. CVPR 2009

[8] David J. Crandall, Lars Backstrom, Daniel Huttenlocher, and Jon Kleinberg. Mapping the World's Photos. WWW 2009

[9] Page, L., Brin, S., Motwani, R., and Winograd, T. The Pagerank Citation Ranking: Bringing Order to the web. technical report, Stanford University, Stanford, CA, 1998

[10] Barrow, H., Tenenbaum, J., Bolles, R., and, Wolf, H. Parametric correspondence and chamfer matching, IJCAI 1977

[11] P.F. Felzenszwalb AND D.P. Huttenlocher. Distance transforms of sampled functions. Technical report, Cornell, 2004