

Tourist Spot Recommendation Applying Generic Object Recognition with Travel Photos

Risa Kitamura
Ochanomizu University
Tokyo, Japan
risak@itolab.is.ocha.ac.jp

Takayuki Itoh
Ochanomizu University
Tokyo, Japan
itot@itolab.is.ocha.ac.jp

Abstract—Thanks to the recent spread of smartphones, tablets and digital cameras, people can take photos easily anytime, and anywhere. Therefore, photos can be a tool to record life logs. We can estimate patterns of actions and movements of people by analyzing their photos. Based on this discussion, we are developing a technique to recommend tourist spots based on the estimation of users’ preferences of traveling plans from their past personal travel photos by using generic object recognition. We applied a generic object recognition system to acquire keywords of subject information taken in the photos and represented the co-occurrence of the keywords by a graph visualization technique. This paper presents our travel recommendation technique and a visual user interface that represents a graph with travel photos based on our graph visualization technique. This paper also introduces our case study with real travel photos.

Index Terms—tourist spot recommendation, generic object recognition, user interface

I. INTRODUCTION

We can take photos anytime and anywhere thanks to the recent evolution of smartphones, tablets, and digital cameras. Consequently, the number of photos taken by ordinary people is increasing. As well as photos of special events such as travel, birthdays, anniversaries, we also take many photos as a part of everyday life. Photos are very convenient to record life logs. We can estimate patterns of actions and movements of people by analyzing their photos.

The goal of our study is to recommend tourist spots based on the estimation of users’ preferences of traveling plans from their past personal travel photos by using generic object recognition. The target users of this research are people who took a lot of travel photos in the past and are thinking about visiting specific cities or regions and would like to easily collect its tourist spot information which they would be interested.

Several travel recommendation systems have been designed based on geotagged photos shared on the Web services including Flickr. However, we cannot suppose all photos have GPS location data. Based on this observation, we are now developing a new technique for recommending travel destinations without geotagged photos. This technique acquires keywords of subject information often taken in the photos by the user, such as landscapes and objects by generic object recognition and uses this keyword as a query to gather tourist information at a specified city or area.

It often happens that photographers do not completely remember what kinds of photos they took. We may need to show such photos in the organized structures so that they remember the contents of photos. It is an important issue for favorable travel recommendation applying previous travel photos taken by users themselves. Based on this discussion, we propose a new visual user interface for structuring a large number of travel photos based on subject information and displaying a list of photos for each subject according to the purpose of travel. We applied our graph visualization technique [1] to represent relationships among keywords retrieved by generic object recognition for this user interface. This method organizes travel photos by displaying not only keywords but also photos. When a user selects photos of particular interest on the user interface, this method gathers keywords corresponding to the selected photos, extracts tourist spot information related to these keywords and recommends it to the user. This paper introduces a case study with real travel photos.

II. RELATED WORK

A. Travel Recommendation Using Geotagged photos

There have been several travel recommendation systems using geotagged photos [2]–[5]. Geotagged photos have been common due to the spread of GPS-enabled cameras, smartphones, photo sharing services and applications displaying geotagged photos on the maps. Collections of travel photos of each user sorted by their timestamps can be regarded as personal travel history.

Kurashima et al. [2] proposed a travel route recommendation method that uses geotagged photos on the photo-sharing site Flickr as travel action histories. This method applies models of travel actions based on the assumption that travelers are likely to visit places that are easy to visit from their current location or match the user’s preferences. Cheng et al. [3] proposed a personalized travel recommendation technique based on personal profiles and users’ attributes (e.g. gender, age, and race). Gao et al. [4] proposed a novel automatic landmark ranking method which can automatically recognize and rank the landmarks for travelers. This method utilizes tags and geotagged photos in Flickr and user knowledge from Yahoo Travel Guide. Cao et al. [5] also proposed a tourism recommendation system based on geotagged photos. The system clusters geotagged photos into groups based on

their locations and finds the representative images for each group. In this system, users can input either a photo of the desired scenery or a keyword describing the place of interest as a query, so that the system can suggest destinations which match the interests of users. However, all of these methods are based on the premise of using geotagged photos. Therefore, these methods cannot be applied to non-geotagged photos.

B. Graph Visualization

This study applies our graph visualization technique [1] to represent relationships among keywords retrieved by generic object recognition. The visualization technique aims to represent connections between important nodes and clusters of other nodes. In other words, we aim to separate important nodes from large clusters so that connections of important nodes become more visible. Based on this discussion, the technique features a clustering step which divides the nodes according to the commonality of connected nodes and similarity of feature value vectors. The technique then calculates the distances between arbitrary pairs of clusters according to the number of connecting edges and similarity of feature value vectors, and finally places the clusters based on the distances. We demonstrated this strategy is effective to represent key persons and their friends, such as famous sports players and their fans in social networks.

KeyGraph [6] is another graph visualization technique focused on the visibility of important nodes. It extracts keywords representing the main claims in a document, constructs a graph treating the keywords as nodes, and displays nodes connecting subgraphs emphatically.

C. Image Browser

Many image browsers have already been developed such as Google Photos and Photos. These can organize, search, and view images by shooting locations and times, as well as persons. Bederson et al. [7] proposed PhotoMesa, a zoomable image browser, applying a hierarchical visualization method. This technique consumes hierarchically grouped images and places them into rectangular regions of display space. Gomi et al. [8] proposed CAT (Clustered Album Thumbnails), a visualization technique for browsing clustered images. This method applies hierarchical clustering to images based on keywords of images and pixel values and selects representative images for each cluster. MoireGraph [9] is a radial focus+context graph visualization technique for displaying visual nodes such as images. MIAOW [10] is a personal photograph browser which displays photographs hierarchically clustered based on shooting times, locations and people.

On the contrary, we apply a clustering algorithm to photos based on only their keywords to cluster photos for travel purposes from past travel photos. We also display a list of photos for each subject according to the purposes of travels.

III. IMPLEMENTATION DETAIL

This section presents our implementation of graph visualization and user interface which group photos by keywords

retrieved from travel photos and provide an interactive mechanism so that users can freely select photos. Our implementation consists of four components: keyword retrieval, graph visualization, user interface for selecting purposes of travels, and query of tourist spots. This section describes each of the components.

A. Keyword Retrieval

We use a generic object recognition function of Microsoft's Computer Vision API [11] to recognize subjects taken in past travel photos, and then assign the photos keywords corresponding to the subjects as tags. This function returns information on the visual content in the image such as objects, living beings, scenery, and actions. It also returns a confidence score that represents the confidence of a tag in real numbers between 0 and 1.

B. Graph Visualization

We denote the confidence of the j -th keyword retrieved from the i -th photo as c_{ij} . Also, let the number of keywords m , and the number of photos n . Our implementation constructs the graph consisting the following nodes and edges.

- A node (corresponding to the j -th keyword) has an n -dimensional vector including c_{1j} to c_{nj} .
- An edge connects two nodes if the inner product between their vectors is larger than a user-specified threshold (0.1 in our implementation). It simply denotes that pairs of keywords which co-occur in same photos are connected by edges.

Here, our algorithm attempts to put particular keywords into the same cluster if they often co-occur in the same photos. The algorithm brings clustering results in which high-level concept keywords and their low-level concept keywords belong to the same clusters.

Our implementation of graph visualization calculates positions of nodes by applying a force-directed layout algorithm and Laplacian smoothing. The graph layout result is saved in JSON format. Also, our implementation displays the graph assigning colors of nodes based on the feature vector values of the nodes. In the case of this study, colors of nodes depicted with IDs of photos related to the keywords. Here, we associate the nodes to the photos which take objects corresponding to the nodes. We assign similar colors to the nodes if IDs of associated photos are close. In practical, nodes with keywords for photos taken on the same day are drawn in the similar colors.

C. User interface for selecting purposes of travels

We developed a new visual user interface which displays clusters of photos and keywords based on the graph layout result. It is implemented on the top of cola.js [12] consuming the graph layout result saved as JSON files. cola.js is an open-source JavaScript library for graph drawing that can constrain such as not to overlap nodes as much as possible. It works with other JavaScript-based visual libraries like D3.js, svg.js, and Cytoscape.js. In this implementation, one cluster based

on the visualization result is defined as one node in the user interface. The root node is defined as follows.

- If a cluster is connected to the largest number of edges, it is defined as the “root node”.
- If a cluster is connected to the root node and the total number of edges connected to the cluster is larger than a user-specified threshold, it is defined as a “hub node”.

This technique also displays keywords belonging to the nodes in addition to nodes themselves. Furthermore, the technique selects several photos corresponding to the keywords and connects these photos by edges from each node. This section calls such a photo as “photo node”. On the other hand, the root node is not connected to the photo nodes by edges. Users can find groups of photos classified by the subject information by adding photo nodes.

The root node, hub nodes, and photo nodes corresponding keywords of hub nodes are displayed as an initial state. The nodes other than the root node and the hub nodes are displayed when a user manually specifies interested nodes.

If a node is connected to a large number of edges, it denotes that the keyword corresponding to the node frequently appears in the generic object recognition results. We can treat such keywords as generic words. In other words, we can provide a user interface which is suitable for narrowing down the purposes of travel, by limiting the displayed nodes according to the number of connected edges.

D. Query of tourist spots

This technique uses the Google Places API Web Service [13], which allows users to query for location information either by proximity or a text string. We use a place search which returns a list of places along with summary information about each place. The Google Places API Text Search Service is a service that returns information about a set of places based on a query phrase. For example, “pizza in Tokyo” or “123 Main Street” can be query phrases. The service responds with a list of places matching the query text string.

This method supposes that users manually select nodes or photo nodes of interest using our user interface. A keyword corresponding to the selected node (or a keyword corresponding to the subject of selected photo node) is used as a query word for the tourist spot search.

Here, it is possible that users are not satisfied by tourist information provided by this method depending on the selected keywords. We would like to implement a relevance feedback method so that the method can selectively return preferable tourist information.

E. Example

This section introduces an example of visualization result using real travel photos. We prepared 2,581 photos taken during abroad trips of the photo owner. We retrieved 14,500 keywords from the photos, visualized the relationships among the keywords, and drew a graph with cola.js based on this visualization result.

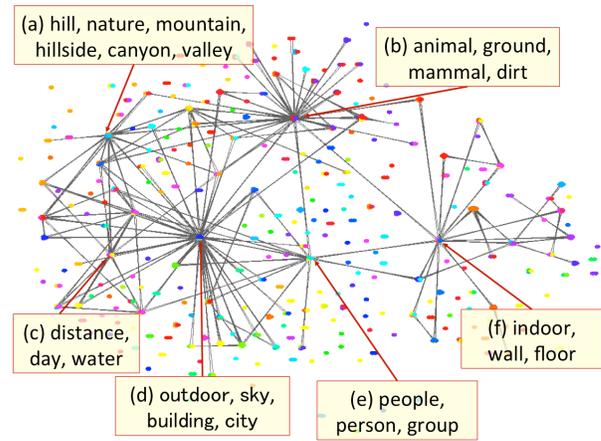


Fig. 1. Hub nodes corresponding to commonly retrieved keywords.

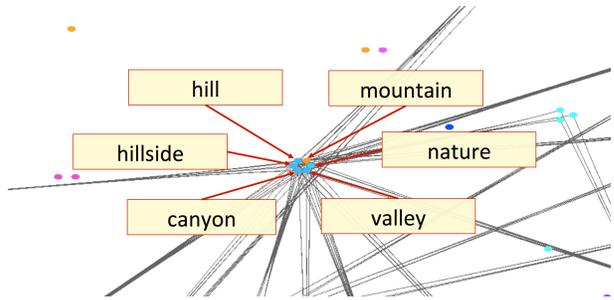


Fig. 2. Enlarged view of hub node (a) shown in Figure 1.

Figure 1 shows how common keywords are visualized as hub nodes in the graph. Figure 2 is an enlarged view of hub node (a) shown in Figure 1, where every node represents one keyword. Edges represent the co-occurrence frequency between clusters and the number of edges to be displayed is adjusted by a user-specified threshold. We suppose these keywords are too general and therefore actually ineffective as query words of information retrieval. We can exclude such keywords belonging to hub node clusters from query words to improve the travel information retrieval results. Or, it is often effective for specific topics to be retrieved by interactively selecting a set of keywords around such common keywords. For example, keywords around (a) in Figure 1 will be helpful to retrieve information about nature and keywords around (b) in Figure 1 will be helpful to retrieve information about zoo or aquarium.

Next, we observed how keywords related to travel or sight-seeing distribute on the display space. Figure 3 shows groups of keywords related specific types of travel or sightseeing. Same categories of keywords appear in the red circles pointed as (a) to (g) in Figure 3. Keywords around Figure 3(a) relate to sightseeing around mountains. Similarly, (b) relates to zoo or aquarium, (c) relates to dining, (d) relates to sightseeing around water-front, (e) relates to traffic, (f) relates to indoor sightseeing, and (g) relates to accommodation or business. This result suggests that this visualization is useful for key-

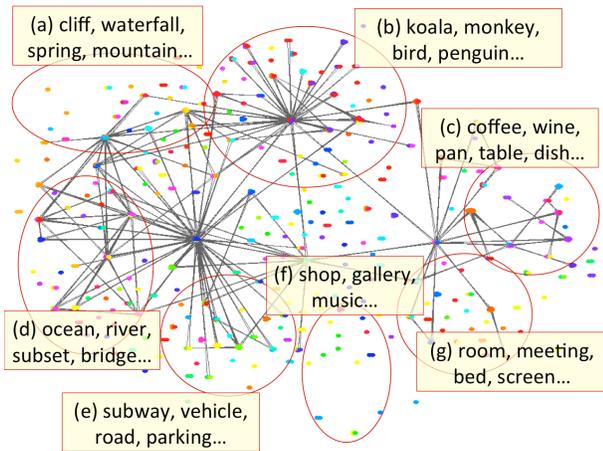


Fig. 3. Keywords.

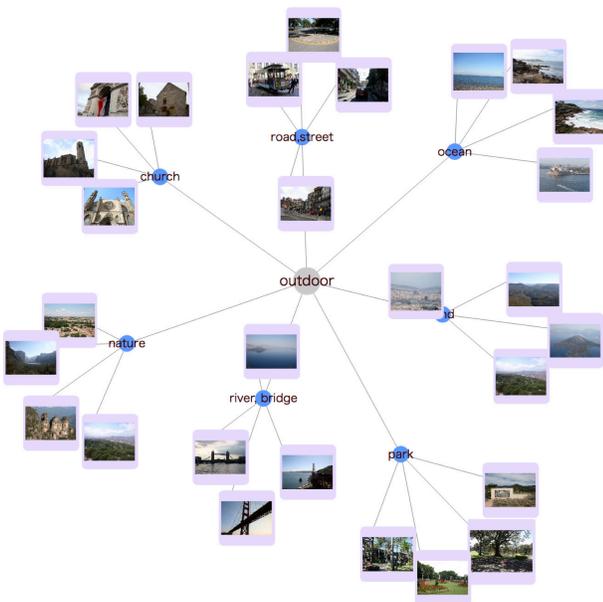


Fig. 4. User Interface.

word selection which improves the retrieval of specific types of travel information.

Finally, Figure 4 shows an example of our new user interface which draws the graph with cola.js. The root node connected to the largest number of edges is placed in the center. The hub nodes which the root node connects by edges have four photo nodes corresponding keywords of each hub node. In our user interface, we assume that the user selects a node or photo node corresponding to a keyword related to the purpose of travel by clicking and the selected keyword is a query for the Google Places API Text Search Requests.

Figure 5 is an example of selected photos and their queries. For instance, when the user is going to travel to Okinawa and selects the picture on the right, "Okinawa", "ocean" and "sea" are queries for the Google Places API Text Search Requests.

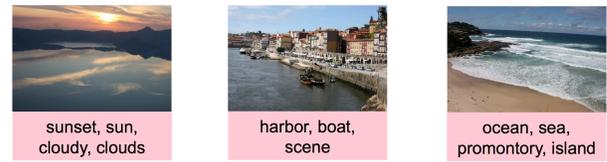


Fig. 5. Example of queries.

We return a list of "ocean" and/or "sea" near Okinawa.

F. Conclusion and Future work

This paper proposed tourist spot recommendation method using subject information of travel photos and the user interface that provides a mechanism to narrow down the purpose of travel interactively. This technique applies a generic object recognition method to travel photos to acquire keywords of subject information taken in the photos. By visualizing the co-occurrence relationship of keywords using our graph visualization technique, we form clusters related to the purpose of travel. In our user interface, a cluster based on the visualization result is defined as one node. We also display the photos corresponding keywords of each node to show what kind of photos the user took in the past and to narrow down the purpose of travel.

We have not completed the implementation including the search function of the tourist spot. Therefore, we would like to implement a function to search tourist spots from queries and develop the user interface. Regarding the keyword, we would like to weight keywords in a cluster to select a query for recommending travel destinations. For example, we would like to calculate weights of keywords from the visualized network connection structure or allow users to select keywords interactively. We would also like to improve the method of selecting photos because the current implementation randomly selects photos corresponding keywords of each node.

ACKNOWLEDGMENT

We appreciate Prof. Kim Marriott of Monash University for helpful suggestions for improvements of our research.

REFERENCES

- [1] T. Itoh, K. Klein, "Key-node-Separated Graph Clustering and Layout for Human Relationship Graph Visualization", IEEE Computer Graphics and Applications, Vol. 35, No. 6, pp. 30-40, 2015.
- [2] T. Kurashima, T. Iwata, G. Irie and K. Fujimura, "Travel route recommendation using geotags in photo sharing sites", In Proceedings of the 19th ACM international conference on Information and knowledge management, pp. 579-588, 2010.
- [3] A.-J. Cheng, Y.-Y. Chen, Y.-T. Huang, W. H. Hsu and H.-Y. M. Liao, "Personalized travel recommendation by mining people attributes from community-contributed photos", In Proceedings of the 19th ACM international conference on Multimedia, pp. 83-92, 2011.
- [4] Y. Gao, J. Tang, R. Hong, Q. Dai, T.-S. Chua and R. Jain, "W2Go: a travel guidance system by automatic landmark ranking", In Proceedings of the 18th ACM international conference on Multimedia, pp. 123-132, 2010.
- [5] L. Cao, J. Luo and A. Gallagher, X. Jin, J. Han and T. S. Huang, "A worldwide tourism recommendation system based on geotagged web photos", IEEE International Conference on Acoustics Speech and Signal Processing, pp. 2274-2277, 2010.

- [6] Y. Ohsawa, N. E. Benson, M. Yachiba, "Key-Graph: Automatic Indexing by Co-occurrence Graph based on Building Construction Metaphor", IEEE International Forum on Research and Technology Advances in Digital Libraries, pp. 12-18, 1998.
- [7] Bederson, B. B., "PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps", Symposium on User Interface Software and Technology, pp. 71-80, 2001.
- [8] A. Gomi, R. Miyazaki, T. Itoh, J. Li, "CAT: A Hierarchical Image Browser Using a Rectangle Packing Technique", 12th International Conference on Information Visualization, pp. 82-87, 2008.
- [9] Jankun-Kelly T. J., Ma K.-L., "MoireGraphs: Radial Focus+Context Visualization and Interaction for Graphs with Visual Nodes", IEEE Symposium on Information Visualization, pp. 59-66, 2003.
- [10] A. Gomi, T. Itoh, "MIAOW: A 3D Image Browser Applying a Location- and Time-Based Hierarchical Data Visualization Technique", Advanced Visual Interface (AVI10), pp. 225-232, 2010.
- [11] Computer Vision API, <https://azure.microsoft.com/ja-jp/services/cognitive-services/computer-vision/>.
- [12] cola.js, <http://marvl.infotech.monash.edu/webcola/>.
- [13] Google Places API Web Service, <https://developers.google.com/places/web-service/intro?hl=ja>.