Mining Tourist Routes from Flickr Photos

Evaggelos Spyrou
Computational Intelligence Laboratory (CIL)
Institute of Informatics and Telecommunications
National Center for Scientific Research-Demokritos
Athens, Greece
E-mail: espyrou@iit.demokritos.gr

Ioannis Sofianos
Department of Informatics and
Computer Technology
Technological Educational Institute
of Central Greece
Lamia, Greece
E-mail: isofianos@teilam.gr

Phivos Mylonas
Department of Informatics
Ionian University
Corfu, Greece
E-mail: fmylonas@ionio.gr

***Abstract*—Popular social networking sites like Flickr are nowadays overwhelmed by geo-tagged photos. Semi-automatic discovery of touristic routes and landmarks from such a pool of photos forms a challenging task. In this paper we attempt to analyze user-generated routes within downtown city areas defined around a pre-selected geographical bounding box and derived from a large geo-tagged Flickr dataset, by utilizing a novel two-level clustering scheme. Our goal is to select routes of touristic interest for a given area. Without loss of generality the latter is considered to be a predefined "window" around a city's most famous landmarks and touristic attractions. The herein proposed framework has been applied to a real-life geo-tagged Flickr photos dataset from a major European metropolis: Athens, Greece.**

I. Introduction

Many popular social networking and photo sharing sites like Flickr[1], Picasa[2], and Panoramio[3], have enabled current multimedia content producers (i.e., every one of us) to pinpoint the geographical locations of their user-generated photos on the map. This functionality resulted to an overwhelming growth of geo-tagged photos on the Web. What makes Flickr special among other social networks is its aspect as an online community, within which users are able to interact with each other. Recently was reported by the technology news and media network "The Verge" that Flickr had a total of 87M registered members and 3.5–10M new photos uploaded daily.

Each Flickr photo may contain useful semantic metadata (tags) added by its photographer that among others describe its geo-location. Combined with the actual visual content and, as well as other metadata descriptions, these wide-spreading geo-tags provide a brand new perspective towards the efficient organization, browsing, and summarization of online multimedia content in general and photo collections in particular. During recent years there has been ever-growing research focus in the recognition and mining of touristic landmarks from such enriched geo-tagged photos. Unlike most existing works in supervised landmark recognition (e.g., [4], [19], [9]), we aim at "recognizing" what routes do tourists actually follow during their visit in a city.

More specifically, in this work we focus on the analysis of user-generated touristic routes within urban areas. Our goal is to select the most representative for a given area of interest. We believe that this research field is going to be trending over the next period, since over the recent years the tourism assistance community is gradually shifting its emphasis to digital, community-based interactive systems focusing on mobile guiding and route recommendations based on social networks.

In principle, planning a travel route is a rather complex and tedious task for people getting to know their destination for the first time. Different sources of information such as travel guides, maps, on-line institutional sites and travel blogs are typically consulted in order to devise the right information that will best cover most subjectively interesting attractions and can be visited within the limited time available in each case. Moreover, a tourist has typically to guess how much time is needed to visit each attraction and to move from one point of interest to the next one. In this manner, aforementioned simple considerations motivate our work herein towards the production of a two-step clustering scheme that may possibly overcome above limitations.

More specifically, to fulfill part of the described goals, we propose a novel method to mine popular touristic routes. We use a large social generated data set derived from Flickr users, we define what we consider as routes, we collect them and upon a clustering scheme we are able to extract a meaningful subset of them. This way, we are able to propose a solution to the first part of the aforementioned problem; a potential traveler may use it, in order to find what similar routes other tourists have followed.

The rest of this paper is organized as follows. In Section II we begin by presenting relevant recent research works. In Section III we describe in detail the approach we propose for the geographic clustering of the collected geotagged photos. Then, in Section IV we describe the approach we followed for gathering and clustering tourist routes. In Section V we present the dataset we have used, implementation issues and experimental results. Finally, results are discussed and conclusions are drawn in Section VI.

II. Related Work

One of the most popular categories of tourist applications focuses on the recommendation of places of interest. In this area, research works aim to automatically discover main attractions, so as to recommend them to their potential users, letting the latter to decide which to visit. In [1] events are

[1]https://www.flickr.com/
[2]https://picasa.google.com/
[3]http://www.panoramio.com/

978-1-4673-8395-0/15/$31.00 ©2015 IEEE

detected by exploiting tags, geo-tags and temporal information. An event is considered to consist of tags with similar temporal and location distributions and also of visually similar photos. In [18] trends and places of interest are detected using a probabilistic approach, taking also into account temporal information about the user. Areas of interest are determined in [7] using geotagged photos and data clustering techniques on spatial and temporal distributions and without any prior knowledge. In [12] events are detected using tags, geotags and temporal metadata of Flickr photos, by first defining event classes, i.e., semantically related groups of events.

Another category of tourist applications extends the former, in the sense that not only main attractions are recommended, but the goal is also to organize the users' schedule and help them visit as many attractions as they wish in a time efficient way. In [13] temporal Flickr metadata are exploited, in an effort to estimate expected visiting times for tourist attractions. This way it is deduced what a tourist is able to visit in a city within a day. In [5] photos around the location of a trip are extracted to create a graph. Then tours that initiate from this location are used to propose one that visits popular places using certain distance constraints, covering all popular landmarks. This work is extended in [14] where spatial and temporal tourist information is mined so as to discover information about trips in popular cities, e.g., what attractions people visit, how long do they stay at each and what are the best panoramic spots.

Similarly, in [2] a system that creates virtual tours and recommends popular places, given a specific regions is presented. In [16] images are spatially clustered, landmarks within them are identified and their popularity is calculated. Recommendations combine minimum distances with maximum tourism popularity. Finally, in [11] an approach for personalization and recommendation of tourist locations is presented. It makes use of users' preferences from their travel history in a city and using this information, it recommends locations in an unknown city.

III. Processing photo geotags

Since a given photo may be geotagged everywhere within the area of interest, we choose to apply a two-level clustering scheme, so as to reduce the number of possible geotags. More specifically, we follow the approach we proposed in [15], where we used this clustering scheme in order to characterize the content in terms of its semantics, spatial and chronological context. In this Section we shall initially briefly present the clustering algorithm we adopted and elaborate on its application on the given dataset.

We should note that when working with Flickr geotagged photos, one should expect that the accuracy of the geotags may be in question, since they are in principle being manually added by the photo owner. However, in some few cases, geotagging is automatically added if an appropriate capturing device (e.g., a camera or a smart-phone) is used. We should emphasize that in the first case, the accuracy depends on the user's knowledge/memory and is subject to errors. However, in the latter case, the accuracy is higher since it depends on the GPS metadata of the capturing device. Having in mind the aforementioned facts and observations, and also considering the study of [3], where the accuracy of Flickr tags is estimated to be at about 11–13m in popular places, we choose to trust the provided geotags, although we expect some "noise". However, due to the clustering procedure, small deviations are exptecte to be filtered.

A. Kernel Vector Quantization

We adopt the *kernel vector quantization* (KVQ) approach of [17] and apply it into two levels: the first produces *geo-clusters*, whereas the second *places*. We begin by summarizing the properties of KVQ and present examples of the resulting geo-clusters after its application on a large Flickr photo dataset. Vector quantization approaches typically use an appropriate set of vectors in order to model probability density functions. Given a set of m data vectors $x_1, x_2, \ldots, x_m \in X$, these are finally represented by a subset of X: $y_1, y_2, \ldots, y_n \in X$, where each x_i is then represented by the nearest y_j, in terms of a pre-defined metric d, with a goal of minimizing a distortion measure. We say that X and d form a metric space, namely (X, d). Moreover, all y_j form a sort of a "codebook" and the aforementioned error is $E = \sum_{i=1}^{m} d(x_i, y(x_i))$, where d may be, e.g., an L_2 metric function and $y(x_i) = \arg\min_{y_i} d(x_i - y_j)$. KVQ tackles this problem using linear programming and upon estimating a sparse solution of the minimization problem, it finally results to a codebook $\mathcal{Q}(D) \subset X$. For the interested reader, further implementation details may be found in [17].

Now, given a point $x \in D$, we begin by defining a *cluster* $\mathcal{C}(x)$ with its center at x as:

$$\mathcal{C}(x) = \{y \in D : d(x, y) < r\}, \tag{1}$$

or in other words as the set of all points $y \in D$ that lie within distance r from x. This distance is the upper bound on distortion, i.e., any two given points in the same cluster are guaranteed to lie "not farther" than r from the cluster center. By applying KVQ on D, we shall obtain the codebook $\mathcal{Q}(D)$, which defines the resulting set of clusters of our interest. Moreover, $\mathcal{Q}(D) \subseteq D$, that is, *codebook vectors* are points of the original dataset. Alternatively, we shall refer to such points as *cluster centers*.

B. Photo Clustering

We apply KVQ on a large set of geo-data. Let P denote the set of available photos. Each photo $p \in P$ is represented as a quadruple, i.e., by its geographic coordinates (*latitude* and *longitude*), the date that it has been taken (according to the metadata) and the user, to whom it belongs. That is, we may consider $p = (lat, lon, date, user)$. In the following, for each of these properties, we shall denote e.g., $p.date$, for improved readability. To apply a first-level of KVQ, it is obvious that we solely need the geographic coordinates. Upon KVQ, we obtain a clustering $\mathcal{C}_1(P) = \{q \in P : d_g(p, q) < r_1\}$. Since these points lie on a sphere (i.e. the Earth), an appropriate metric d_g for this case is the *great circle distance*[4]. It is obvious that for any two given photos $p, q \in P$, $d_g(p, q) = d_g(\langle p.lat, p.lon\rangle, \langle q.lat, q.lon\rangle)$. Let $\mathcal{Q}_1(P)$ denote the set of photos that correspond to the cluster centers of the first level and $\mathcal{C}_1(P) = \{\mathcal{C}_1(p) : p \in \mathcal{Q}_1(p)\}$ denote the cluster collection.

[4]http://en.wikipedia.org/wiki/Great-circle_distance

The second clustering level is a result of KVQ working on each of the aforementioned clusters, separately. In previous work [15], we referred to this semantically enhanced level as “places”. Now, we select a relatively smaller distortion $r_2 : r_2 < r_1$. Let $\mathcal{Q}_2(P)$ denote the set of photos that correspond to the cluster centers of the second level; the cluster collection of this second level $\mathcal{C}_2(P)$ is the union of all second-level cluster collections within each first-level cluster. That is $\mathcal{C}_2(P) = \bigcup_{i=1}^{|\mathcal{Q}_1(P)|} \mathcal{C}_{2,i}(P)$, where the clustering in the i-th second-level cluster is denoted by $\mathcal{C}_{2,i}(P) = \{p, q \in \mathcal{C}_i(P) : d_g(p, q)\}$, where $\mathcal{C}_i(P)$ is the i-th geocluster.

In Fig. 1 we illustrate an example of the application of KVQ in the whole dataset, while in Fig. 2 we depict an example of the application of KVQ in a given cluster. One should notice the density of photos in the city center in the first and particularly in the area of the *Acropolis*, in the latter.

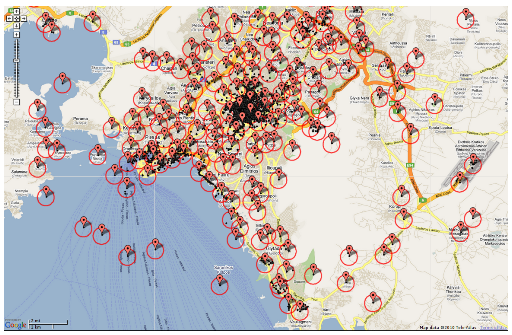


Fig. 1. A map of Athens depicting all geo-clusters. By (black) dots, (red) markers and (red) circles we mark photos, geo-cluster centers and geo-cluster boundaries, respectively.

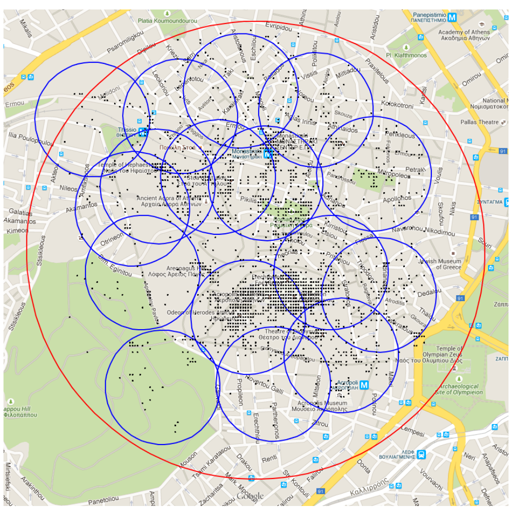


Fig. 2. A geo-cluster (red circle) in Athens' city center and all places (blue circles) extracted within it, using KVQ. Black dots corresponds to photos. The radius of a geo-cluster utilized is $r_g = 700m$, while the one of a place is $r_p = 120m$.

IV. Route Selection

In this Section we present in detail the method we followed for selecting the tourist routes. First we define what a tourist route is, considering our approach, then we discuss the metric we adopted for comparing two routes and finally we describe the clustering procedure we followed.

A. Route Construction

We define a route r as a sequence $\{p_1, p_2, \ldots p_N\}$ of photos. The route members should comply to the following rules:

- $p_1.date \leq p_2.date \leq \ldots \leq p_N.date$, i.e., all members should be consecutive;
- $p_1.user = p_2.user = \ldots = p_N.user$, i.e., all members should belong to the same users;
- let p_i, p_j denote a pair of consecutive route members. Then $p_i.date < p_j.date + T_h$, i.e., a restriction is imposed on the maximum time difference between consecutive photos in order to be considered as consecutive.

Finally, each member of the route is quantized to its nearest second-level cluster center. Thus, the final route r_c is a sequence of clusters, i.e, $r_c = \{\mathcal{C}_{2,i}(P)\}_{i=1}^{N}$.

B. Clustering Distance Function

In order to calculate a means of similarity between two routes, we adopt the Levenshtein distance [8], denoted by d_L in the following. In principle, the Levenshtein distance between two words is defined as the minimum number of edits needed to transform one word into the other, with the allowable edit operations being *insertion*, *deletion*, or *substitution* of a single character. In our case, the role of words is given to routes, thus the role of characters to route members. One interesting property of the Levenshtein distance, is that its lower bound is the difference of the sizes of the two routes and its upper bound is the size of the largest route. Moreover, it is zero if and only if the routes are identical.

C. Hierarchical Clustering

Having computed all distances between routes, the next step is to select an appropriate clustering algorithm. A typical centroid-based algorithm such as K-means may not be applied, since the actual data points are required, something not applicable in our case. Thus, we choose a *hierarchical clustering* algorithm [6] on the set of routes, where solely a distance matrix is needed. This way, the general structure of this algorithm, adjusted for the problem at hand, is as follows:

1) Turn each route into a *singleton*, i.e. into a cluster of a single element.
2) For each pair of routes r_i, r_j calculate the Levenshtein distance $d_L(r_i, r_j)$.
3) Merge the pair of routes that have the minimum $d_L(r_i, r_j)$.
4) Continue at step 2, until a termination criterion is satisfied. The termination criterion most commonly used is the definition of a threshold for the value of the best compatibility indicator.

The clustering procedure creates an agglomerative hierarchical cluster tree, from which clusters occur upon trimming the tree at a given height. The choice of height will provide a partitioning clustering at a certain precision. This way we are able to choose between, e.g., a coarser clustering or a finer one, with the first providing a small number of large clusters and the latter providing a larger number of smaller clusters.

V. Experiments

For the experimental evaluation of our approach we used an urban image dataset which consists of a total of $18,355$ images from the city of Athens. These photos are geo-tagged, dated between 2004–2009 and collected from Flickr using its public API[5]. More specifically we queried Flickr for a region covering the whole city of Athens and retrieved all geo-tagged photos. Then, we manually filtered the dataset to exclude "non-touristic" photos, e.g., commercial, cartoons, etc.. The application of KVQ on the set of geo-tags with radius $r_g = 700$m, as described in Section III-B, resulted to a set of 193 geo-clusters. KVQ was then applied on each geo-cluster, with radius $r_p = 120$m. This resulted to a set of 1546 places.

For the construction of the routes we ensued the following procedure: We first queried the dataset for each user separately. Then we ordered all photos per user at ascending date. As we mentioned in Section IV-A, we set $T_h = 3$h, under the assumption that consecutive photos taken at larger intervals, belong to different routes. We ended up with this conclusion upon careful inspection of the available data set. We then discarded routes that consisted of less than 2 or more than 25 members. We feel that the former do not significantly contribute, while the latter act as noise. In this manner we ended up with 101 routes depicted in Fig. 3.

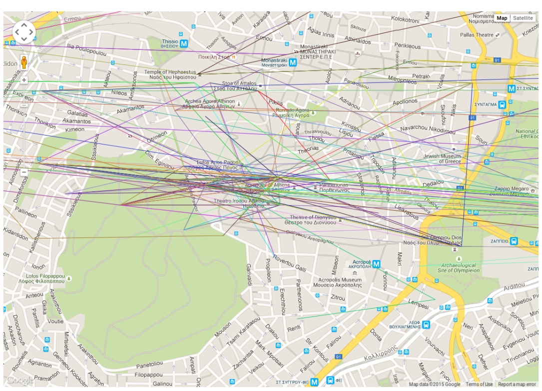


Fig. 3. All 101 routes with more than 2 or less than 25 members. Each route is colored using a random color.

Using the agglomerative hierarchical cluster tree and by trimming at different heights, we are allowed to extract a set of mined tourist routes, as the medoids of the cluster centers. We used the distance criterion in order to merge clusters. This criterion calculates the distance between the two sub-nodes merged at a node to measure node height. All leaves at or below a node with height less than C_d, i.e., its value, are grouped into a cluster. Setting $C_d = 0.6, 0.65, 0.7, 0.75$, we mined 21, 17, 14 and 12 routes, respectively. We illustrate them in Figs. 4(a)–4(d), for the same part of the map depicted in Fig.3. As expected from the quantitative point of view, we may observe that in principle an increased C_d value results into a decreased amount of total clusters. Still, from the qualitative point of view, higher C_d values might provide more meaningful clustering results.

[5]https://www.flickr.com/services/api/

	$C_d = 0.6$	$C_d = 0.65$	$C_d = 0.7$	$C_d = 0.75$
Question a.	82.1%	85.5%	88.0%	92.3%
Question b.	70.0%	64.0%	62.0%	60.0%

TABLE I. User evaluation results.

In order to provide a qualitative evaluation of the proposed approach and since the goal of this work is to assist potential tourists to discover popular and/or interesting routes within a city, we choose to focus on user satisfaction. More specifically, the evaluation scenario involves users[6]. that are to a great extend familiar to Athens city center by being current or former local residents. Our experiment was to present each user the set of the extracted routes and ask the following questions:

a. Which routes do you find relevant to the tourist scenario?

b. Do you feel that the set of the extracted routes is adequate for the tourist scenario?

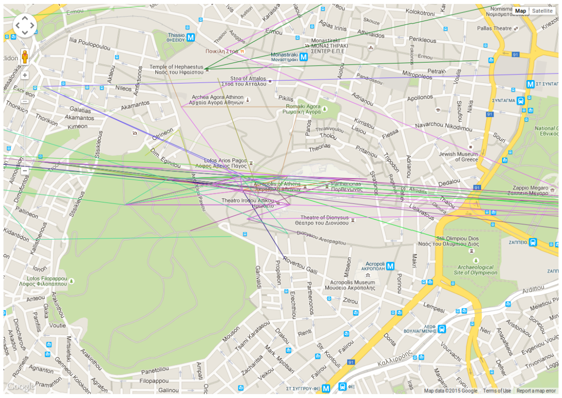
The results are summarized in Table I, where we depict the percentages of the relevant routes and Upon the evaluation and some discussion with the users, it turned out that users were in general satisfied from the extracted routes, however they felt that some areas were not covered adequately. The latter may be explained, since the dataset used contains photos between 2004–2009. However, in the meantime, a few new "hot-spots" have emerged. We also feel that in general, evaluation of tasks aiming at users' satisfaction is known to be a difficult and expensive task, which may involve empirical issues in the process [10]. Nevertheless, to the best of our knowledge there does not exist a dataset with tourist routes, thus it is not feasible to provide some kind of a quantitative evaluation.

VI. Conclusions and Discussion

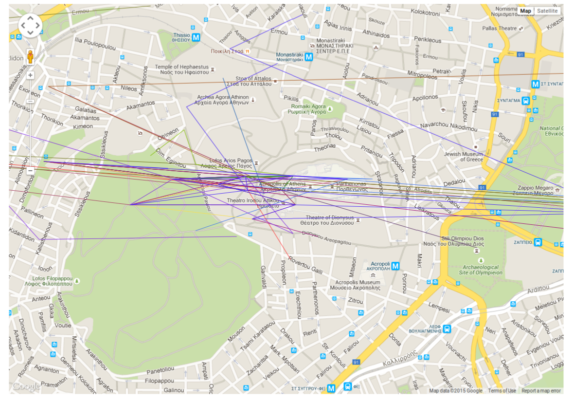
In this paper we briefly presented a novel approach for mining tourist routes, over a large, user-generated data set from Flickr. We showed another aspect of how social networks are able to generate knowledge. An empirical evaluation of our approach from users that have been Athens' residents, indicated its potential and allow us to assume that real users shall also be satisfied.

Future work will focus on the extension of this algorithm and on its application on a significantly larger dataset, from the same geographical area, whose photos span over a period of more than 10 years. We also plan to modify the distance function between routes, so that it will take into account the proximity of two geo-clusters. Another interesting aspect would be the inclusion of the time intervals between two consecutive route members, which shall allow us to extract information related to the interestingness of specific spots. Finally, we feel that we should perform a qualitative evaluation with real users, to assist us on further improvements.

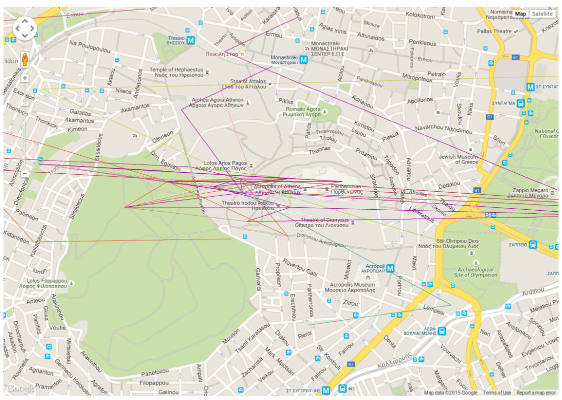
[6]More specifically users derived from two (2) academic institutions:
a. Technological Educational Institute of Central Greece, 28 students
b. Ionian University, 22 students



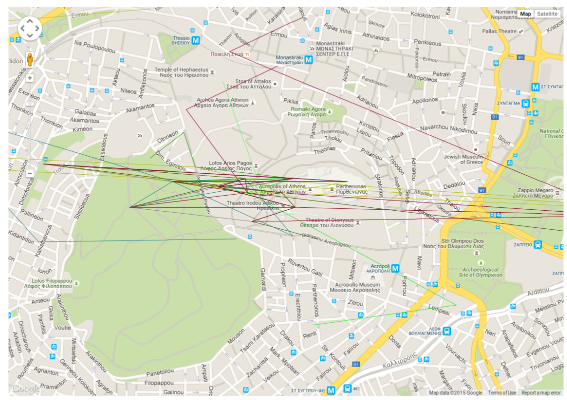
(a) $C_d = 0.6$ resulted to 21 clusters.



(b) $C_d = 0.65$ resulted to 17 clusters.



(c) $C_d = 0.7$ resulted to 14 clusters.



(d) $C_d = 0.75$ resulted to 12 clusters.

Fig. 4. Extracted sets of mined touristic routes for various values of the C_d parameter.

Acknowledgment

The authors would like to thank Giorgos Siantikos for his valuable suggestions that helped to significantly improve this paper.

References

[1] L. Chen and A. Roy, *Event detection from flickr data through wavelet-based spatial analysis*. In Proc. of ACM CIKM, 2009.

[2] Q. Hao, R. Cai, J.-M. Yang, R. Xiao, L. Liu, S. Wang and L. Zhang, *Travelscope: standing on the shoulders of dedicated travelers*. In Proc. of ACM MM, 2009.

[3] C. Hauff, *A study on the accuracy of Flickr's geotag data*. In Proc. of ACM SIGIR, 2013.

[4] J. Hays and A.A. Efros, *IM2GPS: estimating geographic information from a single image*. In Proc. of IEEE CVPR, 2008.

[5] S. Jain, S. Seufert and S. Bedathur, *Antourage: Mining Distance-Constrained Trips from Flickr*. In Proc. of ACM WWW, 2010.

[6] L. Kaufman, and P.J. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009.

[7] S. Kisilevich, D. Keim, N. Andrienko and G. Andrienko, *Towards Acquisition of Semantics of Places and Events by Multi-perspective Analysis of Geotagged Photo Collections*. Geospatial Visualisation, pp. 211–233, Springer, 2013.

[8] V. I. Levenshtein, *Binary codes capable of correcting deletions, insertions, and reversals*, Soviet Physics Doklady 10, pp. 707–10, 1966.

[9] Y. Li, D.J. Crandall, and D.P. Huttenlocher, *Landmark classification in large-scale image collections*. In Proc. of IEEE ICCV, 2009.

[10] Y. Liu, J. Bian, E. Agichtein, *Predicting information seeker satisfaction in community question answering*, In SIGIR '08 Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 483–490, 2008.

[11] A. Majid, L. Chen, Ling, G. Chen, H.T. Mirza, I. Hussain and J. Woodward, *A context-aware personalized travel recommendation system based on geotagged social media data mining*. Int. Journal of Geographical Information Science, vol.27, no.4, pp. 662–684, Taylor & Francis, 2013.

[12] N. Nitta, Y. Kumihashi, T. Kato and N. Babaguchi, *Real-World Event Detection Using Flickr Images*. MultiMedia Modeling, pp. 307–314, Springer, 2014.

[13] A. Popescu and G. Grefenstette, *Deducing trip related information from flickr*. In Proc. of ACM WWW, 2009.

[14] A. Popescu, G. Grefenstette and P.-A. Moëllic, *Mining tourist information from user-supplied collections*. In Proc. of ACM CIKM, 2009.

[15] E. Spyrou and Ph. Mylonas, *Analyzing Flickr metadata to extract location-based information and semantically organize its photo content* Neurocomputing, Elsevier, article in press, doi:10.1016/j.neucom.2014.12.104

[16] Y. Sun, H. Fan, M. Bakillah and A. Zipf, *Road-based travel recommendation using geo-tagged images*. Computers, Environment and Urban Systems, Elsevier, 2013.

[17] M. Tipping and B. Schölkopf, *A Kernel Approach for Vector Quantization with Guaranteed Distortion Bounds*, Artificial Intelligence and Statistics, pp. 129–134, 2001.

[18] S. Van Canneyt, S. Schockaert, O. Van Laere and B. Dhoedt, Bart, *Time-dependent recommendation of tourist attractions using Flickr*. In Proc. of BNAIC, 2011.

[19] Y. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T. -S. Chua, and H. Neven, *Tour the world: Building a web-scale landmark recognition engine*, CVPR, 2009.