# Detecting the long-tail of Points of Interest in tagged photo collections


Christos Zigkolis
Aristotle University
54124, Thessaloniki, Greece
chzigkol@csd.auth.gr

Symeon Papadopoulos
Informatics & Telematics Inst.
57001, Thessaloniki, Greece
papadop@iti.gr

Yiannis Kompatsiaris
Informatics & Telematics Inst.
57001, Thessaloniki, Greece
ikom@iti.gr

Athena Vakali
Aristotle University
54124, Thessaloniki, Greece
avakali@csd.auth.gr



## Abstract

*The paper tackles the problem of matching the photos of a tagged photo collection to a list of "long-tail" Points Of Interest (PoIs), that is PoIs that are not very popular and thus not well represented in the photo collection. Despite the significance of improving "long-tail" PoI photo retrieval for travel applications, most landmark detection methods to date have been tested on very popular landmarks. In this paper, we conduct a thorough empirical analysis comparing four baseline matching methods that rely on photo metadata, three variants of an approach that uses cluster analysis in order to discover PoI-related photo clusters, and a real-world retrieval mechanism (Flickr search) on a set of less popular PoIs.*

*A user-based evaluation of the aforementioned methods is conducted on a Flickr photo collection of over* $100,000$ *photos from* $10$ *well-known touristic destinations in Greece. A set of 104 "long-tail" PoIs is collected for these destinations from Wikipedia, Wikimapia and OpenStreetMap. The results demonstrate that two of the baseline methods outperform Flickr search in terms of precision and F-measure, whereas two of the cluster-based methods outperform it in terms of recall and PoI coverage. We consider the results of this study valuable for enhancing the indexing of pictorial content in social media sites.*


## 1. Introduction

The massive amounts of user contributed content in social media sites has provided valuable input for a series of mining applications and for numerous intelligent services built on top of the mined knowledge. An important problem that has recently attracted considerable interest is the detection of Points of Interest (PoIs) in large user-contributed photo collections [5, 10, 3]. It has been demonstrated that photos of popular landmarks around the world can be successfully mined in large sets of tagged photos. In addition, different representative photos for each detected landmark can be identified [5, 11] leading to more diverse pictorial descriptions for landmarks. The results of automatic PoI detection are valuable in the context of tourist applications both for presenting the interesting attractions of a place to the potential visitor [11, 3] as well as for planning efficient tourist itineraries based on the available PoIs [2, 4, 9].

Despite the great interest in this problem, most existing works are limited to the discovery of prominent landmarks that are well represented in the photo collections. Less known PoIs are usually disregarded due to the fact that most existing methods require a large number of photos per PoI in order to reliably detect it. The objective of this paper is to evaluate the effectiveness of different schemes for identifying photos that depict "long-tail" PoIs. Starting from three geographical sources of information, namely Wikipedia[1], Wikimapia[2], and OpenStreetMap[3], we compile lists of "long-tail" PoIs. Then, we devise several matching functions in order to associate individual photos with PoIs by use of textual (title, description, tags) and geolocation metadata. In addition, we leverage the graph-based photo clustering framework of [8] to discover photo clusters related to the target PoIs. Three variants of the method are tested that rely on different types of photo similarity graphs (visual, tag-based, and hybrid). Finally, we make use of the Flickr search service by posting queries constructed from the PoI name and place. The latter implementation is a widely used real-world retrieval mechanism, and hence, constitutes a suitable and high performance competi-

[1] Wikipedia, http://www.wikipedia.org/
[2] Wikimapia, http://wikimapia.org/
[3] OpenStreetMap, http://www.openstreetmap.org/

tor. Our empirical study demonstrates that two of our baseline matching schemes outperform Flickr search in terms of F-measure, while the graph-based methods perform best in terms of recall and coverage, thus being more suitable for the problem of "long-tail" PoIs, where conventional matching schemes may yield no matches.

The rest of the paper is structured as follows. Section 2 briefly discusses several related works. In Section 3, we present background information on the problem, namely the necessary notation, the PoI list compilation process, and the employed graph-based photo clustering method. Section 4 presents the proposed matching methods for addressing the problem. Section 5 describes the experimental setup and the obtained results. Section 6 concludes the paper and discusses future work.

## 2. Related Work

The application of social media mining in tourist scenarios has recently attracted significant interest due to its potential for the automatic production of high-quality tourist-related multimedia content. For instance, Kennedy and Naaman [5] make use of tags, location information and visual features of photos in order to identify clusters of photos that correspond to different views of popular landmarks. However, they make no use of external knowledge sources (e.g. Wikipedia) and they rely on a large number of geotagged and representative photos for each landmark. Thus, their approach is limited to landmarks that are well covered (in terms of pictorial content and metadata) within a photo collection. Similar limitations hold for the work in [11] that relies on the "interestingness" property of photos (provided by Flickr), which limits its utility to highly voted photos.

Quack et al. [10] mine landmarks and events from a large set of photos by clustering them based on their visual and textual similarity, classifying the photo clusters into landmarks or events and mapping the clusters to Wikipedia articles by use of query formulation and visual matching. Despite the high precision reported by the authors, their method also suffers from low recall, i.e. it does not detect places with few photos in Flickr.

Other travel-oriented applications of social media mining are presented in [1, 2, 4, 9]. Crandall et al. [1] attempt to estimate the geographic position of tagged photos by use of visual and textual features, which is complementary to our work, since the more geotagged pictures are available the better the performance of the PoI detection will be. The works in [2, 4, 9] deal with the automatic travel itinerary creation from tagged photo collections of cities. Such applications can greatly benefit from PoI detection, since they rely on PoIs to create itineraries passing through them. The work in [9] makes use of Wikipedia articles and categories in order to identify PoI names and locations.

## 3. Background - Notation

### 3.1. Notation

Table 1 contains all necessary notation used throughout the paper. Our starting point is the collection (set) $\mathbb{R} = \{r\}$ of tagged photos, where each photo $r$ is a tuple $(t_r, d_r, X_r, \lambda_r)$, comprising a title $t_r$, a description $d_r$, a set of tags $X_r$ and the capture location of the photo $\lambda_r$, expressed as a latitude-longitude pair of values. Furthermore, we consider the set $\mathbb{P} = \{p\}$ of PoIs, where each PoI $p$ is a tuple $(pl_p, T_p, \lambda_p)$ containing the place of the PoI $pl_p$, a set $T_p$ of alternative titles for the PoI and the location $\lambda_p$ of the PoI. The problem we address is the evaluation of different mappings $f : \mathbb{R} \rightarrow \{\mathbb{P}, nil\}$, which map each photo of the collection to one of the available PoIs or to no PoI at all. In the case of cluster-based PoI-photo matching (described in subsection 4.2), we achieve the PoI-photo mapping $f$ through an intermediate mapping $f : CL_{type} \rightarrow \{\mathbb{P}, nil\}$, which associates each photo cluster of the cluster set $CL_{type}$ to one of the available PoIs or to no PoI at all. The photo clusters are derived based on the graph-based photo clustering method described in subsection 3.3. Three types of clusterings are considered depending on the underlying image similarity graph; more specifically $type \in \{VIS, TAG, HYB\}$ corresponding to visual, tag-based, and hybrid similarity graph respectively. A cluster $c \in CL_{type}$ is a tuple $(M_c, RT_c)$, where $M_c$ is a set of cluster members (i.e. $M_c \subset \mathbb{R}$) and $RT_c$ is the set of representative titles for the given cluster, derived from the process described in subsection 4.2.

### 3.2. PoI list creation

We use three sources for compiling the list of PoIs: (a) Wikipedia, (b) Wikimapia and (c) OpenStreetMap. The advantage of Wikipedia is that it contains rich additional information for each PoI contained in it. However, it misses several less important PoIs. Wikimapia and OpenStreetMap, on the other hand, contain only basic information for each PoI, but they have much higher coverage of PoIs, especially in smaller places and cities. We populate our final list having in mind that we are more interested in "long-tail" PoIs than popular ones. Starting the selection from a list of touristic destinations, we end up with a number of PoIs ready to be used by the PoI-photo matching methods.

### 3.3. Graph-based clustering

The cluster-based PoI-photo matching method of subsection 4.2 relies on the creation of three types of photo similarity graphs representing three kinds of similarities between photos of the collection, namely visual, tag and hybrid. In

**Table 1. Notation used in the paper**

| Symbol | Definition |
|---|---|
| $\mathbb{R} = \{r\}$ | Collection of photos |
| $\mathbb{P} = \{p\}$ | Set of PoIs |
| $CL_{type} = \{c\}$<br>$type$ | Set of clusters<br>*VIS, TAG or HYB* |
| $r = (t_r, d_r, X_r, \lambda_r)$<br>$t_r$<br>$d_r$<br>$X_r$<br>$\lambda_r$ | Photo metadata<br>title<br>description<br>tags<br>geolocation |
| $p = (pl_p, T_p, \lambda_p)$<br>$pl_p$<br>$T_p$<br>$\lambda_p$ | PoI information<br>place<br>representative titles<br>geolocation |
| $c = (M_c, RT_c)$<br>$M_c$<br>$RT_c$ | Cluster metadata<br>members $M_c \subset \mathbb{R}$<br>titles $RT_c = \{rt_{c_1}, rt_{c_2}, \ldots\}$ |
| $tok(s)$ | set of tokens for string $s$ |
| $place(p)$ | alternate names for place of $p$ |

the visual graph, the edge weights represent the pairwise similarities in terms of visual content (SIFT descriptors [7] are extracted for each photo and a bag-of-visual-worlds feature vector is computed based on the software implementation of [12]). On the other hand, the tag graph is built by use of tag co-occurrences between photos. Each edge on this graph is weighted by the number of tags shared between the two photos. Both popular tags and weak edges are discarded to increase noise resilience and reduce the computational needs of the clustering algorithm. The hybrid graph is formed by considering the union of the visual and tag graph. Then, a community detection procedure is applied on each graph with the goal of identifying sets of nodes (i.e. photo clusters) that are more densely connected to each other than to the rest of the network. This graph-based clustering framework is described in detail in [8].

# 4. PoI-Photo Matching

## 4.1. Baseline matching

In order to match Flickr photos with the extracted PoIs, we make use of the following metadata: (a) title, (b) tags, (c) description, and (d) capture position. Due to the unrestricted nature of photo sharing applications, there are numerous cases, in which one or more of the aforementioned metadata fields are missing. In such cases, it is not possible to match the photo to any PoI by use of the baseline methods proposed here. We consider matching functions of the form $s : \mathbb{R} \times \mathbb{P} \rightarrow [0, 1]$ for estimating how well a particular photo $r$ matches a given PoI $p$. In particular, the matching functions can be expressed as:

$$s_k(r,p) = \begin{cases} A_k & \text{if matching criterion } \phi_k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $\phi_T, \phi_D, \phi_L$ and $\phi_H$ are the criteria for title-tag, description, location and hybrid matchings respectively.

In order to simplify the function specification, we also define the tokenization function $tok(s)$, which, given a string $s$, produces a set of tokens (strings) by splitting $s$ around matches of white spaces. We further define the function $place(p)$, which, given a PoI, returns the name $pl_p$ of the place containing the PoI, and some alternate names for it, e.g. $place(\text{“Acropolis”}) = \{\text{“Athens”}, \text{“Athina”}\}$. Finally, we slightly abuse the subset operator to denote string containment, i.e. $a \subseteq b$ denotes that string $a$ is contained in string $b$. Depending on the kind of metadata we rely on, we obtain a different matching criterion $\phi$ and a different score $A$. In the end, we get the following matching function variants:

- Title+tag token match ($s_T$): $A_T = \alpha$, $\phi_T \equiv \{\exists t \in T_p :$ $(tok(t) - place(p)) \subseteq (tok(t_r) \cup X_r)\}$
- Location match ($s_L$): $A_L = \beta$, $\phi_L \equiv \{d(\lambda_r, \lambda_p) < l\}$ where $d(\lambda_r, \lambda_p)$ is the geodesic distance between $\lambda_r$ and $\lambda_p$, and $l$ is a predefined threshold.
- Description match ($s_D$): $A_D = \gamma$, $\phi_D \equiv \{\exists t \in T_p : t \subseteq d_r\}$
- Hybrid match ($s_H$): $A_H = 1$, $\phi_H = \phi_T \cap \phi_L$

The positive scores returned by the aforementioned variants obey the constraints $\gamma < \beta < \alpha < 1$ in order to reflect the confidence we have in each criterion. In the end, for each criterion $k$, given PoI $p$, we obtain the set $POI_k(p) = r : s_k(r,p) > 0$ of photos matched with $p$ by use of matching criterion $k$.

## 4.2. Cluster-based matching

Here, we attempt to match a PoI $p$ with a cluster $c$ by using the PoI and cluster titles, $T_p$ and $RT_c$ respectively. The latter is a result of a process that finds the most frequent word sequences within the titles of photos of the cluster $c$. More specifically, we define the function $seq(t_r)$ that, for each title $t_r$ of a photo, returns the set of all possible term sequences up to length 6. We then aggregate over all term sequence sets for the photos of the cluster maintaining a count for each one of them. In the end, we select the top five term sequences as the representative titles $RT_c$ of the cluster, by ranking them with a function that takes into account the term sequence frequency, the sequence length (we prefer longer to shorter titles), and the lexical diversity of titles (we prefer to have diverse titles in order to capture alternative names).

After extracting $RT_c$, we apply the cluster-based matching procedure that is implemented as a cascade of three consecutive criteria, namely full title match ($\phi_F$), relaxed title match ($\phi_{Rel}$), and partial lexical match ($\phi_{Par}$). We consider matching functions of the form $s_{CL} : \mathbb{CL} \times \mathbb{P} \rightarrow [0,1]$ for estimating how well a particular cluster $c$ matches a given PoI $p$. The matching $s_{CL}$ is expressed by Equation 2.

$$s_{CL}(c,p) = \begin{cases} \mu & \text{if matching criterion } \phi_F \\ \nu & \text{if matching criterion } \phi_{Rel} \\ \xi & \text{if matching criterion } \phi_{Par} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $\phi_F$, $\phi_{Rel}$, and $\phi_{Par}$ are defined as:

- $\psi_F \equiv \{\exists t \in T_p : t \subseteq RT_c\}$
- $\psi_{Rel} \equiv \{\exists t \in T_p : (tok(t) - place(p)) \subseteq tok(RT_c)\}$
- $\psi_{Par} \equiv \{\exists t_1 \in T_p, \exists t_2 \in RT_c :$ $|\{(pt_i, ct_j)|LV(pt_i, ct_j) > \sigma_1\}| > \sigma_2 * |PT|\}$

  where $LV(s_1, s_2) \in [0,1]$ denotes the Levenshtein similarity [6] between the strings $s_1$,$s_2$, $pt_i \in PT = \{(tok(t_1) - place(p))\}$ and $ct_i \in \{(tok(t_2) - place(p))\}$ and $\sigma_1 = 0.6, \sigma_2 = 0.5$.

The full title match criterion ($\phi_F$) searches for identical titles, while the relaxed criterion ($\phi_{Rel}$) tries to match all tokens of a PoI title, regardless of their sequence, with tokens from a cluster title taking into account the function $place(p)$. The third criterion ($\phi_{Par}$) tries to match titles, of which a certain number of tokens is lexically quite similar. With the partial lexical match we are trying to solve both the problems of misspelling and the identification of slightly different written words with the same meaning (e.g. "Saint Anna" or "St. Anna"). We set the values of the constants in Equation 2, such that the following condition holds: $\mu > \nu > \xi > \sigma_2$. In the end, given PoI $p$, we obtain the set $POI_{CL}(p) = \bigcup M_c, c : s_{CL}(c,p) > \sigma_2$ of photos matched with $p$ by use of cluster-based matching.

### 4.3. Flickr search matching

For a PoI $p$, we form a query $q_p$ by concatenating the PoI name (most characteristic name from the set $T_p$) and the place name $pl_p$ (in order to avoid ambiguity for landmarks with the same name in different places). We post the query to the Flickr search API (flickr.photos.search) and then obtain the set $POI_{FL}(p) = FLICKR(q_p)$ of the photos matched with $p$ by use of Flickr search matching.

## 5. Experiments

### 5.1. Dataset

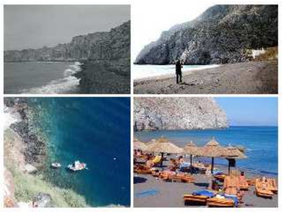
Starting from 10 touristic places of Greece [4], we compiled a list of 104 "long-tail" PoIs contained in them. Furthermore, we downloaded a number of place-focused photos (129.023 in total, 46.181 geotagged) by issuing appropriate queries for each place to the Flickr search service (e.g. "Chania Greece" to retrieve photos from "Chania"). This collection will be referred to as the "original" photo collection and be denoted as $\mathbb{R}_0$. The use of the Flickr search matching method ($FL$) on the list of PoIs resulted in an extension of the "original" dataset, since Flickr search is conducted over the whole Flickr collection. Adding these additional photos to our "original" collection resulted in a total of $\backsim$ 148.000, which will be referred to as the "extended" photo collection, and denoted as $\mathbb{R}_{ext}$.

The application of the proposed matching methods took place separately for each place, which means that PoIs of a place were matched with photos (or clusters) of the same place. For instance, for each PoI of Santorini, for which 20,086 photos were downloaded from Flickr, the candidate photos for matching were limited to the set of 20,086 photos, or to the 187 photo clusters in the case of the hybrid graph cluster-based matching ($CL_{HYB}$).

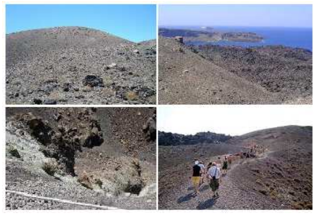
### 5.2. User study

We conducted a study involving 10 users. Each user was assigned approximately 20 PoIs, and for each PoI, she was presented with the union of the results of all matching methods for that PoI, the so-called "PoI photo pool". The user was instructed to decide for each photo whether it was relevant or irrelevant to the PoI in question. If the user could not decide, they were given the option of not filling in the annotation for the photo (*don't know*). In order to help the users make the decision, for each PoI we provided a link to a site containing a characteristic photo and some description of the PoI. In total, a set of $\backsim$ 34.000 photos (the union of all PoI photo pools for the 104 PoIs of the study) were evaluated. Each photo was evaluated by two users in order to be able to estimate inter-annotator agreement ($\kappa$-statistic). Although the users were oblivious to the matching method that produced the photos for a given PoI, this association was maintained in the back-end, thus enabling us to compute precision ($P$), recall ($R$), and F-measure ($F$) for each one of the methods. Furthermore, since methods could not identify relevant photos for each of the 104 PoIs, we computed the PoI coverage ($C$) for each method, i.e. the percentage of PoIs, for which at least one relevant photo could be retrieved by the method.

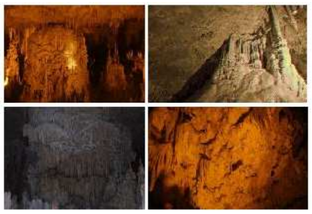
[4] Chania, Corfu, Heraklion, Ioannina, Nafplion, Naxos, Paros, Santorini, Thessaloniki, Zakynthos



(a) Red Beach (Santorini)



(b) Nea Kameni (Santorini)



(c) Perama Caves (Ioannina)

(d) Patsides (Herakleion)

**Figure 1. Examples of long tail PoIs.**

**Table 2. IR performance results (relaxed, $\mathbb{R}_0$)**

| **Method** | $P$ | $R$ | $F$ | $\kappa$ | $C$ |
|---|---|---|---|---|---|
| $T$ | 0.680 | 0.466 | **0.553** | 0.252 | 0.894 |
| $D$ | 0.613 | 0.176 | 0.273 | 0.160 | 0.760 |
| $L$ | 0.533 | 0.271 | 0.359 | -0.478 | 0.490 |
| $H$ | **0.753** | 0.069 | 0.126 | **0.498** | 0.270 |
| $CL_{HYB}$ | 0.525 | 0.429 | 0.472 | -0.464 | **0.952** |
| $CL_{TAG}$ | 0.518 | **0.515** | 0.516 | -0.419 | **0.952** |
| $CL_{VIS}$ | 0.626 | 0.107 | 0.183 | -0.020 | 0.450 |
| $FL$ | 0.677 | 0.437 | 0.531 | 0.263 | 0.800 |

## 5.3. Results and discussion

Depending on whether we considered the "don't know" response as relevant or irrelevant, we computed two different retrieval performance sets, which we will refer to as *relaxed* ("don't know" counts as relevant) and *strict* ("don't know" counts as irrelevant). In addition, depending on the photo set used to compute the retrieval performance scores, we obtained two different results, one computed with reference to $\mathbb{R}_0$ and the other with reference to $\mathbb{R}_{ext}$. The combination of the aforementioned variants of computing performance resulted in four different result sets, namely relaxed in $\mathbb{R}_0$ (Table 2), strict in $\mathbb{R}_0$ (Table 3), relaxed in $\mathbb{R}_{ext}$ (Table 4) and, strict in $\mathbb{R}_{ext}$ (Table 5).

Tables 2 and 3 present valid comparative results, since all matching methods are evaluated with reference to the same photo collection. One can see that the highest precision was achieved by use of the hybrid ($H$) matching method ($75.3\%$ in relaxed, and $68\%$ in strict mode). Inter-annotator agreement was also high ($0.498$). However, this method gave the lowest recall values due to the added criterion of location in the matching procedure. In our dataset, most of the photos

**Table 3. IR performance results (strict, $\mathbb{R}_0$)**

| **Method** | $P$ | $R$ | $F$ | $\kappa$ | $C$ |
|---|---|---|---|---|---|
| $T$ | 0.591 | 0.485 | **0.533** | 0.027 | 0.894 |
| $D$ | 0.534 | 0.189 | 0.279 | -0.176 | 0.760 |
| $L$ | 0.439 | 0.262 | 0.328 | -2.003 | 0.490 |
| $H$ | **0.676** | 0.083 | 0.147 | **0.400** | 0.270 |
| $CL_{HYB}$ | 0.442 | 0.435 | 0.438 | -0.942 | **0.952** |
| $CL_{TAG}$ | 0.432 | **0.514** | 0.470 | -0.710 | **0.952** |
| $CL_{VIS}$ | 0.553 | 0.112 | 0.186 | -0.493 | 0.450 |
| $FL$ | 0.600 | 0.458 | 0.520 | -0.016 | 0.800 |

do not include geographic information, thus, this method is not applicable for them.

Considering recall values, we see that the tag cluster-based method $CL_{TAG}$ achieved the best score ($51.5\%$ in relaxed, $51.4\%$ in strict mode), which indicates that tag-based photo clustering can be beneficial for extending the PoI search results. However, this comes at a cost in precision, since the $CL_{TAG}$ method presents the worst performance in terms of precision. The highest $F$-measure was achieved by the title+tag ($T$) method ($55.3\%$ in relaxed, and $53.3\%$ in strict mode) making it the best candidate for balanced results in terms of precision and recall. An additional noteworthy observation pertains to the very low $\kappa$-statistic scores achieved by methods (with the exception of the highly selective Hybrid ($H$) matching method). This indicates that associating photos with PoIs is a hard task even for human annotators. For instance, some users may find relevant only the characteristic views of a PoI while others may also find relevant some alternative views of it (e.g. indoor views). In addition, some PoIs are intrinsically difficult to evalutate (e.g. neighborhoods, beaches, old cities).

Finally, the best PoI coverage was achieved by two of the cluster-based methods, $CL_{HYB}$ and $CL_{TAG}$, since they managed to find relevant photos for 99 out of the 104 PoIs of the list. Figure 1 presents four examples of PoIs, for which Flickr could not return any relevant photo, while the proposed methods could match several related ones. We also computed for each method the number of PoIs, for which the method returned the maximum number of relevant photos. In the case of the "strict" annotation mode, the title+tag matching ($T$) returned the most relevant photos for 34 PoIs, description ($D$) for 1, location ($L$) for 9, hybrid graph clustering ($CL_{HYB}$) for 12, tag graph clustering ($CL_{TAG}$) for 26, and Flickr search matching ($FL$) for 22.

Although Tables 4 and 5 are not valid for comparison, they can be used to draw the following interesting conclusion: even when the Flickr search matching uses the whole Flickr dataset, the performance of the method does not improve in terms of coverage. This marks the importance of cluster-based matching methods in enriching "long-tail"

**Table 4. IR performance results (relaxed, $\mathbb{R}_{ext}$)**

| Method | $P$ | $R$ | $F$ | $\kappa$ | $C$ |
|---|---|---|---|---|---|
| $T$ | 0.680 | 0.345 | 0.458 | 0.252 | 0.894 |
| $D$ | 0.613 | 0.128 | 0.212 | 0.160 | 0.760 |
| $L$ | 0.533 | 0.214 | 0.305 | -0.478 | 0.490 |
| $H$ | **0.753** | 0.044 | 0.083 | **0.498** | 0.270 |
| $CL_{HYB}$ | 0.525 | 0.349 | 0.420 | -0.464 | **0.952** |
| $CL_{TAG}$ | 0.518 | 0.410 | 0.458 | -0.419 | **0.952** |
| $CL_{VIS}$ | 0.626 | 0.084 | 0.147 | -0.020 | 0.450 |
| $FL$ | 0.651 | **0.526** | **0.582** | 0.201 | 0.800 |

**Table 5. IR performance results (strict, $\mathbb{R}_{ext}$)**

| Method | $P$ | $R$ | $F$ | $\kappa$ | $C$ |
|---|---|---|---|---|---|
| $T$ | 0.591 | 0.358 | 0.446 | 0.027 | 0.894 |
| $D$ | 0.534 | 0.140 | 0.222 | -0.176 | 0.760 |
| $L$ | 0.439 | 0.203 | 0.277 | -2.003 | 0.490 |
| $H$ | **0.676** | 0.051 | 0.094 | **0.400** | 0.270 |
| $CL_{HYB}$ | 0.442 | 0.353 | 0.392 | -0.942 | **0.952** |
| $CL_{TAG}$ | 0.432 | 0.407 | 0.419 | -0.710 | **0.952** |
| $CL_{VIS}$ | 0.553 | 0.087 | 0.151 | -0.493 | 0.450 |
| $FL$ | 0.573 | **0.544** | **0.558** | -0.071 | 0.800 |

PoIs with photos, and thus in improving the search experience delivered by search systems based on them.

## 6. Conclusions and Future Work

The paper proposed a set of methods for dealing with the challenging problem of matching Points of Interest to photos. Four baseline methods which make use of photo metadata were used to directly associate photos with the available PoIs. In addition, we exploited a photo clustering algorithm by associating the derived clusters with the available PoIs. All proposed methods were experimentally evaluated and compared with the results of the Flickr search service. The results demonstrate improved performance for some of the proposed methods compared to Flickr search. In particular, the hybrid matching method presents higher precision rate surpassing Flickr search by $11\%$. This method also resulted in the highest inter-annotator agreement. In terms of recall, the tag cluster-based matching surpassed Flickr search matching by $17\%$, while in terms of $F$-measure the title+tag matching method scored highest achieving an improvement of $6\%$ compared to Flickr search matching. Finally, the tag and hybrid cluster-based methods resulted in the highest coverage surpassing Flickr search matching by $15\%$. In summary, for popular PoIs, for which there is abundant photographic content on the Web, it is most appropriate to use high precision matching methods such as the hybrid ($H$) and the title+tag ($T$) matching methods. In contrast, for "long-tail" PoIs, for which conventional matching schemes cannot find any match, the proposed cluster-based matching methods ($CL_{HYB}$ and $CL_{TAG}$) are more appropriate.

In the future, we intend to investigate two ways for further improving the results of PoI-photo matching (a) exploit PoI categories (e.g. museum, church, castle) in order to build optimized category-specific PoI-photo matching strategies, (b) investigate the potential of ensemble methods to increase performance.

## 7. Acknowledgments

This work was supported by FP7 projects WeKnowIt (215453) and Glocal (248984), co-funded by the EC. Mr. Christos Zigkolis' work was funded by the Irakleitos II national project.

## References

[1] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *WWW '09*, pages 761–770, New York, NY, USA, 2009. ACM.

[2] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, and C. Yu. Constructing travel itineraries from tagged geo-temporal breadcrumbs. In *WWW '10*, pages 1083–1084, New York, NY, USA, 2010. ACM.

[3] Y.-T. Z. et al. Tour the world: building a web-scale landmark recognition engine. In *Proceedings of International Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, June, 2009.

[4] S. Jain, S. Seufert, and S. Bedathur. Antourage: mining distance-constrained trips from flickr. In *WWW '10*, pages 1121–1122, New York, NY, USA, 2010. ACM.

[5] L. S. Kennedy and M. Naaman. Generating diverse and representative image search results for landmarks. In *WWW '08*, pages 297–306, NY, USA, 2008. ACM.

[6] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. Technical Report 8, 1966.

[7] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[8] S. Papadopoulos, C. Zigkolis, Y. Kompatsiaris, and A. Vakali. Cluster-based landmark and event detection on tagged photo collections. *Multimedia, IEEE*, 2010.

[9] A. Popescu, G. Grefenstette, and P.-A. Moëllic. Mining tourist information from user-supplied collections. In *CIKM '09*, pages 1713–1716, New York, NY, USA, 2009. ACM.

[10] T. Quack, B. Leibe, and L. Van Gool. World-scale mining of objects and events from community photo collections. In *CIVR '08*, pages 47–56, New York, NY, USA, 2008. ACM.

[11] Y. Ren, M. Yu, X.-J. Wang, L. Zhang, and W.-Y. Ma. Diversifying landmark image search results by learning interested views from community photos. In *WWW '10*, pages 1289–1292, New York, NY, USA, 2010. ACM.

[12] K. E. van de Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Trans. on Pattern Analysis and Mach. Intelligence*, 99, 2009.