Learning Landmarks by Exploiting Social Media

Chia-Kai Liang, Yu-Ting Hsieh, Tien-Jung Chuang,
Yin Wang, Ming-Fang Weng, and Yung-Yu Chuang*

National Taiwan University

Abstract. This paper introduces methods for automatic annotation of landmark photographs via learning textual tags and visual features of landmarks from landmark photographs that are appropriately location-tagged from social media. By analyzing spatial distributions of text tags from Flickr's geotagged photos, we identify thousands of tags that likely refer to landmarks. Further verification by utilizing Wikipedia articles filters out non-landmark tags. Association analysis is used to find the containment relationship between landmark tags and other geographic names, thus forming a geographic hierarchy. Photographs relevant to each landmark tag were retrieved from Flickr and distinctive visual features were extracted from them. The results form ontology for landmarks, including their names, equivalent names, geographic hierarchy, and visual features. We also propose an efficient indexing method for content-based landmark search. The resultant ontology could be used in tag suggestion and content-relevant re-ranking.

1 Introduction

As digital cameras and storage get cheaper, many of us have thousands of photographs in our own albums. Their effective management becomes increasingly important but more difficult nevertheless. *Image annotations* have been shown effective to facilitate organization and retrieval of photograph collections. However, automatic image annotation algorithms for generic semantic are still far from being applicable. A good news is that automatically collected metadata, such as time and location, and their derived information have been proved helpful in management of photo collections [1]. Almost all digital cameras record *time stamps* when pictures were taken. Some *location-aware cameras* can augment location information about where pictures were taken by using GPS, cellular or Wi-Fi Networks. Unfortunately, although automatically-collected location context is useful, location-aware cameras do not grow at a rapid rate because of cost, power consumption and image quality. Thus, most images still lack of geographic metadata for effective organization and retrieval.

Even though it is useful to automatically add geographic tags to photographs taken by non-location-aware cameras, limited by available content analysis technology, it is still hopeless to automatically annotate general geographic names for photographs in the near future. Thus, this paper focuses on *landmark photographs*, pictures of a specific but useful category. Figure 1 shows the overview of our system. There are two phases in our system, the *pre-processing phase* and the *application phase*. In the pre-processing phase, we downloaded from Flickr a total of 11,028,186 *geotagged* photos

* This work was supported by grants NSC97-2622-E-002-010-CC2 and NTU-98R0062-04.

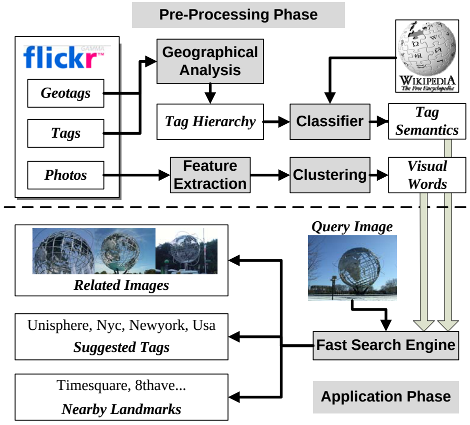
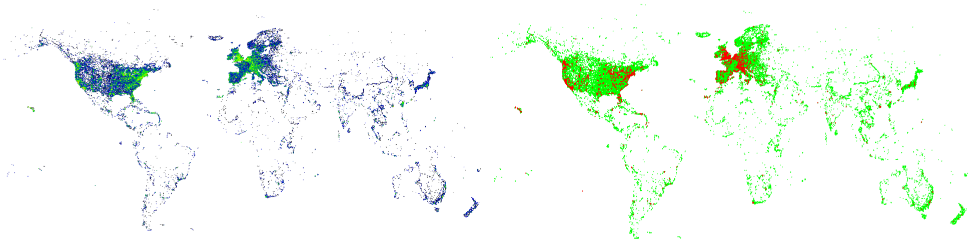


Fig. 1. Overview of the proposed system.

which were uploaded by 140,948 users during 2005/01/01 and 2008/01/01. Geotags record latitude and longtitude coordinates where the pictures were taken. They are either recorded by cameras or labeled by users, and may contain errors. Figure 2(a) shows the spatial distributions of all these retrieved photographs over the world. We perform statistical analysis on geotags to find textual tags with strong spatial coherence. These tags more likely refer to geographic terms. Landmark tags are further classified from these geographic tags using corresponding Wikipedia articles. In addition, association analysis is used to build a *tag hierarchy*, from which we can find the containment and equivalence relationships among geographic terms. Section 3 describes our methods for extracting the above information. In Section 4, for each landmark tag, Flickr is queried to retrieve a collection of relevant images. From each set of images, we extract a set of distinctive features associated to a specific landmark tag. Hence, we have a database containing a set of landmark tags and each tag has its own set of *visual words*. For efficient indexing, these visual words are clustered into a hierarchical tree.

With the learnt landmark ontology (landmark names, synonyms, hierarchy, visual words and so on), many interesting applications become possible. The application phase utilizes the ample and precise data extracted from the pre-processing phase for various applications. For example, after the user uploads a new photo for query, our system can immediately identify the landmark in the photo. It can then suggest the name of the landmark, return the related or representative images of the landmark, recommend other proper tags for this image, and even show the user the nearby landmarks within the same city. Although similar to Kennedy and Naaman's work [2], our paper can be taken as a further step by providing the following improvements:

- It puts more focus on landmarks and proposes better methods to identify landmarks' textual tags while they simply borrowed the idea from Ahern *et al.*'s paper [3]. Along this line, this paper introduces tag hierarchy construction by association analysis and landmark classification using corresponding Wikipedia articles. Thus, we can extract more structured information, such as synonyms of landmarks and their containment relationships, to form a more complete landmark ontology.



(a) (b)

Fig. 2. Spatial distributions of the retrieved photos and landmark tags. (a) The spatial distribution for geotags. There are totally 11,028,186 geotagged photos in our database mirrored from Flickr. A warmer color represents locations with more photos and a colder color means the ones with less photos. (b) The spatial distribution of landmark tags. We have identified a total of 3,821 landmark tags over the world. These tags are shown in red. Green pixels represents the coverage of all geotags.

- Their work mainly focus on finding representative images, but this paper proposes methods for more efficient landmark search by visual contents.
- They only demonstrated their method within San Francisco area. On the contrary, this paper explores the world's landmarks. In addition, this paper demonstrates a set of interesting applications enabled by the discovered landmark ontology.

2 Related work

In recent years, there are quite a few work on exploring the usage of geographic metadata. Toyama *et al.* described WWMX, a system for browsing geo-referenced photo collections [4] and various issues related to alike systems. Mor Naaman and his colleagues have done lots of exciting work on various topics related to geo-referenced tags and photographs, including tag visualization [5, 3], extracting the event and place semantics [6, 7], and ranking representative images [8, 2]. The main differences distinguishing our work from theirs are that most of them assume existence of geographic metadata and confine the usage to the geotagged images. On the contrary, we perform visual analysis to tag images and use the hierarchical visual words to avoid the expensive pair-wise similarity measurement, making the system more scalable and robust.

University of Oxford conducted a series of researches on applying the text search techniques to the image search problem [9–11]. The key idea is to treat the local distinct features, or *visual words*, in an image as the words in a document. However, in their approach, each image in the dataset are considered as an unique entity and therefore a large index storage is required. On the contrary, in our approach, hundreds or thousands of images from a single landmark are automatically grouped together. Their visual word distributions are aggregated into a single one. Therefore, our system can be easily scaled to deal with millions of on-line images. Also our system provides novel applications beyond simply finding similar images, such as automatic tag suggestion. Finally, Hays and Efros use global image features to match an image to the geotagged images [12]. However, this method is not accurate enough for serious applications.

3 Textual Tags for Landmarks

The geotags reveal the geographic distribution of a tag, which consequently can be used as the signal for landmarks. For example, the distributions of "birthday" and "beach" are wide and sparse on Earth, but that of "Statue of Liberty" is local and clustered. We exploit this property to identify geographically related tags. Furthermore, we build a tag hierarchy of those tags according to their co-occurrence and geographical distributions. However, pure geographical analysis is not sufficient to identify landmarks precisely. Tag distributions of either cities or villages are sometimes similar to that of landmarks. One of the reasons is that photos are not uniformly distributed in those areas, and might be clustered at some specific places. Therefore, differences between landmark and non-landmark tags are sometimes not distinguishable from spatial distributions. In order to tell whether a tag is a landmark, we use knowledge stored in Wikipedia to build a classification model.

3.1 Geographical Analysis

A photo P_i has the following attributes relevant to our application: (1) geotags $L_i = (x_i, y_i)$, (2) photographer u_i and (3) tag set $T_i = \{t_{i_1}, ..., t_{i_{n_i}}\}$ where n_i is the number of tags associated with P_i. To identify which tags are geographic terms, photographs' geographic locations are grouped by tag names to form clusters. Specifically, the geographic cluster of a tag t is $C_t = \{L_j | t \in T_j\}$. If a geographic cluster of a tag is localized in a small region, this tag is likely to refer to a place. One issue is that photos with geographic tags are not always at the right place due to labeling errors or other reasons. These photos are considered as *noises*, which are handled with RANSAC. For each tag t, we first create its geographic cluster C_t with size $|C_t|$. We randomly pick up one point $\mathbf{x}$ from C_t and use it as the center of a Gaussian. The deviation σ is determined by taking the $(68\% \times |C_t|)$-th closest point (68% is confidence level of Gaussian model in 1σ). The fitness of the hypothetic Gaussian is evaluated as $\sum_i G(L_i; \mathbf{x}, \sigma)$, where $L_i \in C_t$. The process is repeated several times and the Gaussian $G_t(\mathbf{x}, \sigma)$ with the best fitness is chosen to describe the spatial distribution of the geographic cluster.

After deciding $G_t(\mathbf{x}, \sigma)$, we collect photos located within 3σ as inliers. The area $A(t)$ of the tag t is defined as area of the convex hull of all inliers. Because non-geographic tags tend to be distributed wildly over the world, we keep the tags whose areas are smaller than a threshold and send them to the tag hierarchy construction stage in the next section. Among 60,449 tags that go through geographic analysis, 13,854 of them are identified as geographic terms.

3.2 Tag Hierarchy

A photo can have multiple tags, which are usually semantically or geographically related. Because we have eliminated non-geographic tags in the previous step, the remaining co-occurred tags are very likely geographically related. For example, photos labelled with "Statue of Liberty" are often labelled with "New York" as well, but "Golden Gate Bridge" is not likely to appear with "Africa". We use association analysis to formulate their closeness. Assuming $P(b|a)$ denotes the probability that photos labelled with

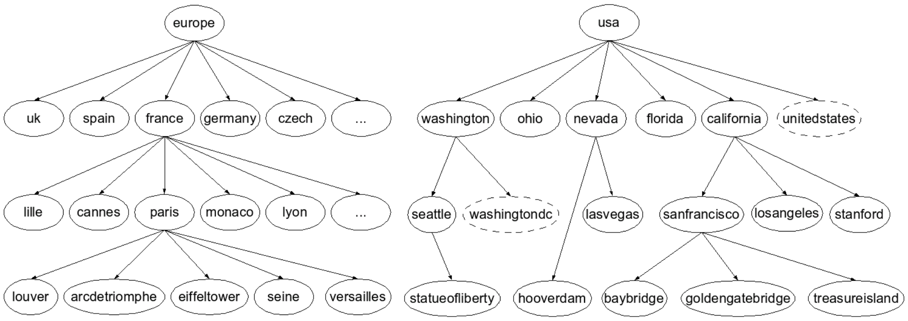


Fig. 3. Examples from the tag hierarchy. The synonyms are shown in dotted circles.

tag b given that tag a is labelled; $N(a, b)$ denotes the number of photos with both tags a and b; and $N(a)$ the number of photos with tag a, we have $P(b|a) = \frac{P(a \cap b)}{P(a)} = \frac{N(a,b)}{N(a)}$. The most related tag to tag a, $M(a)$, is defined as $M(a) = \arg\max_b P(b|a)$. Given a tag a, if we iteratively evaluate $M(a), M(M(a)), \cdots$, eventually it will reach a tag like "USA," "Europe," "Asia," "Africa," etc. A sequence of tags generated in this way is called a *trace*. An example trace beginning from `spaceneedle` is shown as follows:

$$\begin{aligned}
&M(\texttt{spaceneedle}) = \texttt{seattle} && \text{with } P = 0.959924 \\
&M(\texttt{seattle}) = \texttt{washington} && \text{with } P = 0.294886 \\
&M(\texttt{washington}) = \texttt{usa} && \text{with } P = 0.0914492
\end{aligned}$$

Some interesting properties can be observed in the traces. First, the synonyms are usually the most related tag to each other; i.e., $M(M(a)) = a$. Second, a tag which is the ancestor of other tags is less likely to be a landmark tag. It usually corresponds to a district or an even larger area. We create the trace of each geographic tag independently and then merge them into a tag hierarchy. Two examples in Figure 3 show the subsets of the tag hierarchy. We can clearly see the hierarchical relationships between the tags, from continents of root nodes to individual landmarks of leaf nodes.

3.3 Wikipedia Knowledge

The leaf nodes of the tag hierarchy may still not correspond to a landmark. It could be a local event or an unattractive static object. We show that the exact semantics of the tag can be inferred from the corresponding article in Wikipedia, and thus the accuracy of the landmark identification can be further improved. For each tag, we find the corresponding article on Wikipedia. Note that The synonyms for a landmark are already merged in the tag hierarchy. Thus, for a group of synonyms, we only use the one with the highest count. Among 13,854 tags which pass geographic analysis, less than 10 tags did not have a Wikipedia article. In these cases, we classify them as the class of "other".

Inspired by the spam detection algorithms, we formulate our problem as a classification problem. Each article should belong to exactly one of three classes *landmark*, *city*, and *others*. The *city* class contains not only the city-scale tags, but also all the areas that are larger than a specific attractive, natural or man-made structure, such as districts, towns, and beaches. The *other* class contains all other things, including local events or even non-geographical elements. We find that this three-class formulation can significantly improve the accuracy. We add the class of *city* because the articles in the *city* class contain several unique descriptions (population, etc) and therefore they should not be mixed with the ones in the *other* class. In addition, identifying the names of those large areas would potentially provide novel applications.

We use the occurrence of the tokens (words of the "wiki text") as the features of the article to perform classification. Because the articles on Wikipedia are relatively terser and more precise than general documents, the naïve-Bayesian model can provide fairly accurate results. Let $P(W|C)$ denote probability that the word W appears in the documents of class C and $P(C|A)$ denote the probability that document A belongs to the class C. Using Bayesian rule, we have

$$P(C|A) = P(C|T_1, ..., T_n) = \frac{p(C) * p(T_1, ..., T_n|C)}{p(T_1, ..., T_n)}, \tag{1}$$

where T_i is the i-th token in the document. The most likely class of A would be $c^*_A = \arg\max_c P(C = c|A)$. In addition to the individual words, using n-gram as tokens can significantly improve the accuracy. This is because that the Wiki article uses many deterministic sentences and formal keywords when describing cities and landmarks.

To verify the performance of our classifier, we manually label 634 tags as the ground truth for validation. For building the classifier, we randomly choose 50 tags for each class as the training data. The accuracy of three-class classification using single words is 78.9% and improved to 86.1% when 2-gram and 3-gram are included as tokens.

3.4 Results and discussions

To summarize, there are a total of 2,068,833 distinct textual tags we retrieved from 11,028,186 geotagged photos of Flickr. Among them, 60,449 tags were used by more than 15 users. After geographic analysis, 13,854 of them are considered geography-related. Among them, 4,633 tags are classified as landmarks by the wiki-article classifier (with an accuracy around 85%). Considering the tag hierarchy, 3,821 of them appear at the leaf and are considered landmark tags. Figure 2(b) shows the spatial distribution of these landmark tags. Note that the distribution of landmarks are biased to Flickr users' patterns. As an example, below are some landmarks we identified within London area.

```
greenwich bigben londoneye waterloo docklands battersea kew (kewgardens) canarywharf
tate (tatemodern) westminsterabbey towerbridge riverthames londres brixton sciencemuseum
housesofparliament heathrow batterseapowerstation trafalgar (trafalgarsquare) leicester
nationalgallery harrods cuttysark clapham gherkin britishmuseum crystalpalace
```

Tags in parenthesis are synonyms. The off-the-shelf databases could give landmark name as well. However, our approach has the following advantages. First, most of those databases are more interested in administrative hierarchy. Landmarks are often not the main focus. Thus, landmarks are not necessarily listed. Second, even if they are, landmarks could have multiple names, but not all are listed in the off-the-shelf databases. Finally, off-the-shelf database can be outdated, but information extracted from social media keeps updated and reflects how landmarks are really tagged in social media.

Our approach shares a similar goal and part of the methodology as Ahern *et al.*'s world explorer [3]. However, our system has the following features: more emphasis on landmarks and the incorporation of tag hierarchy and Wikipedia-classification. These give better results. Using London as an example, here are the tags that are at the leaf under London but classified as "others" by our wiki-article classifier, `guesswherelondon`, `londonbus`, `londonist` and `londonunderground`. It means that all four have a landmark-scale cluster in the geographic analysis. With only geographic analysis like Section 3.1 and Ahern *et al.* did, they can't be distinguished from real landmarks. In addition, the tf-idf measure used by Ahern *et al.* can find a better tag to represent a group of tags in one area, but it does not change the number of clusters. On the contrary, we use the tag hierarchy to merge the synonyms. Also, two landmarks in a small area are not mixed together in our method. Finally, Ahern *et al.* segmented the earth into many regions in a multi-level pyramid. On the contrary, we perform the analysis globally. This can remove some non-geographical tags such as `baseball` and `soccer`.

4 Visual Features for Landmarks

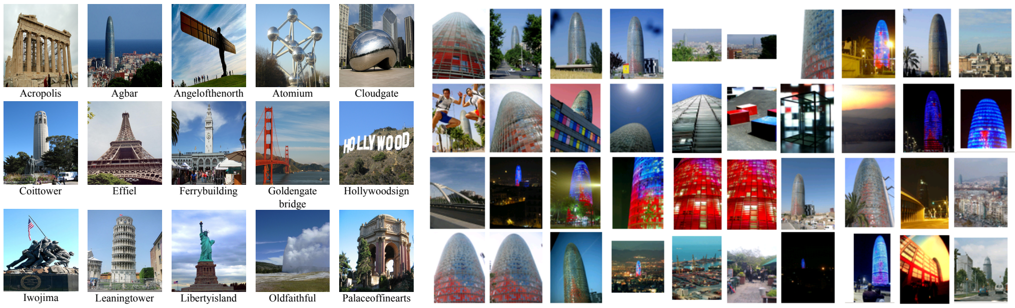
This section shows how to exploit the visual information of the landmark photos (i.e., images with the tags classified as landmarks) for content-based image retrieval. The system must be *robust* and *fast*, returning the results immediately after given the query image. Additionally, system should be *scalable* to handle millions of photos.

4.1 Hierarchical Visual Words Construction

Since many landmarks are made of similar materials and shot under similar illumination conditions, traditional global image features such as color histogram can hardly be used to distinguish one landmark from another. A landmark is recognizable due to its unique structures and thus it is better to use locally distinct features. Here we apply SIFT [13] to detect the interest points in the photo. There are usually hundreds to thousands of SIFT features in a single image and therefore it is impractical to store all features in the database and perform pairwise feature matching to all of them in the query phase. Here we adopt the concept of *visual word* [14]. All features are coarsely quantized into many clusters using k-means and each image can be considered as an article written using those clusters. In this way, many techniques in text retrieval can be readily applied [9, 11]. To recognize thousands of landmarks, we still have to use a large number of clusters to preserve the distinctness. This could significantly slow the matching process. Here we quantize the features in a hierarchical fashion [15]. In the beginning all feature are clustered into k clusters and the features in one cluster are further clustered into k sub-clusters. This process is perform recursively until a specific storage limit is reached. All the leaf nodes are the final visual words.

4.2 Efficient Indexing and Search

In the search phase, features in the query image are detected and each is assigned to the nearest visual word. This can be done very efficiently by traversing the tree using best-first search if the approximate nearest visual word is sufficient. We also use a modified



(a) the 15 landmarks used in the evaluation

(b) a subset of images for `agbar`

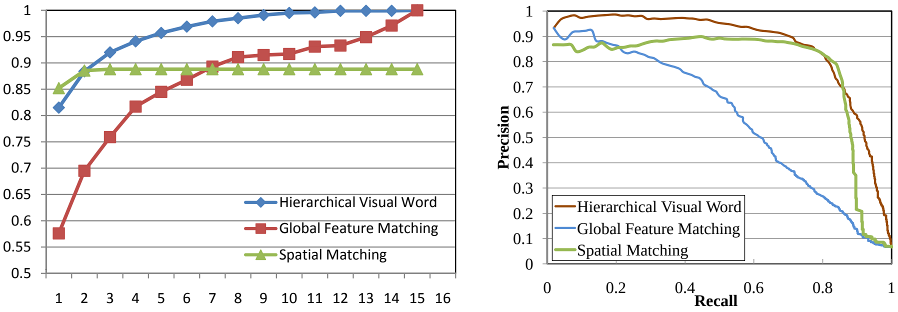
Fig. 4. (a) The 15 landmarks used in the experiments. (b) A subset of images retrieved from Flickr using `agbar` as the keyword. It shows a great deal of visual diversity and contains a few "noises".

n-best search method [16] to improve the accuracy. Instead of only traversing the best path, we traverse the first n best paths in parallel.

In previous methods, each visual word is attached with a backward index to the images containing that word, and the ranking of the retrieved images depends on the characteristics of the indices [10]. However, this approach requires huge storage when there are millions of images to be indexed (in [10], only 5k positive images were indexed). Also, retrieving the very similar images may not be useful in many applications since they give no more information than the original query image. To resolve these problems, we propose to index the landmarks instead of photos. For each visual word, we record the backward index to landmarks containing that word. This method is more useful than the per-image indexing for many reasons. First, the number of landmarks in much fewer than the number of photos, and increase at a much slower rate. Second, together with our tag semantics and geographical analysis, identifying the landmark is enough for many applications. Third, this indexing method is more robust to the noisy tag inputs. Few irrelevant images in the training data would not affect the search results. In terms of the text retrieval, our method attempts to categorize the input article, not to retrieve the similar ones from the database. After this step, other related but not similar articles in that category can be retrieved using other existing techniques. Specifically, at each leaf node v of the hierarchical tree, we store the number of the landmarks that containing the visual words N_v and the number of occurrences in the i-th landmark $C_v(i)$. When a feature is assigned to a the visual word v, we increase the score to the i-th landmark by a modified tf-idf function $C_v(i)log(N/N_v)$, where N is the total number of images in the database.

4.3 Evaluation

The construction of ground truth for the landmark image query is labor-intensive, so we only choose 15 landmarks for evaluation (Figure 4(a)). For each landmark, we manually examined and selected at least 500 Flickr photos that indeed capture the landmark structure. These images naturally covers many different illuminations and view positions. For training, we randomly pick 2,700 images (180 for each landmark) from the



(a) (b)

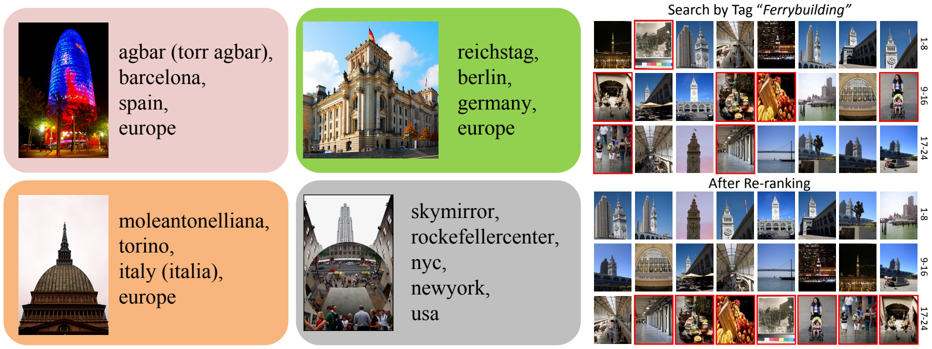
Fig. 5. (a)The average accuracy using three different methods. (b) Their average PR curves.

ground truth to construct the hierarchical tree. There are totally 1,942,243 raw feature vectors. The degree of the clustering at each level is 8, and the maximal tree depth is 11. The final hierarchical tree contains 778,011 visual words. The tree consumes 528MBs and the backward index consumes 17.5MBs.

Two methods are compared. The first one uses the global feature to measure the per-image similarity. (We used six features including histogram, Gabor texture and others; SVM is used for classification.) The second one uses the spatial matching to measure the similarity [10]. (Count the number of feature matches between the query image and each of the images in the database. The feature matches are verified by spatial constraints.) For the first method, the best accuracy of the top return is only 57.6% although it is slightly faster than our algorithm (0.211 seconds). The spatial matching method has a higher accuracy than our method in the first return, but its performance soon saturates after the first 3 returns. It is because many tested images can never find a match image in the database when it is not big enough. Also the spatial matching is much slower than our method. On average, each query takes 54.843 seconds. Figure 5(a) shows the overall performance for three methods. Figure 5(b) shows the PR curves. These show that our method is compared favorably to the other two methods.

For testing robustness, we replace the training data with photos queried from Flickr by tags, which are more convenient to obtain but also noisier. For example, Figure 4(b) shows part of the retrieved images from Flickr for *Torre Agbar*, a 21st-century skyscraper in Spain. It contains many photos without the landmark. Although many of those photos are not visually related to the landmarks, the performance is only decreased by 2%. The degradation can be easily compensated by increasing the number of visual words. This shows that the hierarchical tree combined with the per-landmark indexing is very robust to noise in training data.

Finally, for testing in a larger scale, we increase the number of landmarks to 150. For dataset of this size, we can only use Flickr's returned images as both data for training and ground truth for evaluation. In this case, the average accuracy of the top return is 30%, which is much higher than that of the random guess (0.6%), and again can be increased by increasing the number of visual words. The real performance should be better since the "ground truth" here are actually retrieved using Flickr's search engine



(a) automatic tag suggestion (b) image re-ranking

Fig. 6. Applications using the discovered landmark ontology. (a) Automatic tag suggestion. Once the landmark in the image is classified, the tags in the same trace of the tag hierarchy could be added. The synonyms are listed in parenthesis. (b) The result of the image re-ranking. The top shows the top 24 images returned by Flickr when using `Ferrybuilding` as the keyword. The red-framed images are the obvious outliers. The bottom are the results after visual re-ranking. We can see that, after re-ranking, they have the lowest scores.

containing much noise (Figure 4(b)). This shows that our method is highly scalable and robust. Categorizing more landmarks does not require much labeling effort to build the training data, and the storage only increases linearly with the number of the landmarks.

5 Applications

This section presents a set of applications which uses the built landmark ontology. Other potential applications include *attraction map construction* and *album management*.

Landmark identification from images. As shown in the previous section, our system can identify the presence of a landmark in an image efficiently and accurately. Once we identify this, the landmark tag and its derived tags can be automatically added or more photographs related to this landmark could be displayed depending on the application.

Automatic tag suggestion. Our landmark ontology eases landmark image annotation by borrowing tags learnt from those who tagged photos of the same landmark. Once a landmark is detected, the ontology suggests a set of potential tags. Figure 6(a) gives some results. For example, our system recognizes that the top-left photo of Figure 6(a) contains the *Agbar Tower*. A set of tags are then suggested, `agbar`, `barcelona`, `spain` and `europe`. In addition, `torreagbar` is suggested since the system recognizes that `agbar` and `torr agbar` are both dialects referring to the Agbar Tower by Flickr users.

Visual relevance re-ranking. The ontology can also be used to re-rank results for landmark image search by considering not only textual relevance but also visual content. Figure 6(b) shows an example using the keyword `ferrybuilding`. On the top, we see the top 24 images returned by Flicker. The bottom show the results after re-ranking. We can see that all the irrelevant images now have lower scores. The overall processing time is 4.53 seconds in this example.

6 Conclusion

This paper proposes methods to automatically transfer tags to unlabeled photographs from annotated landmark photographs of a photo-sharing website. We use geographic analysis, tag hierarchy construction and wiki-article classification to identify landmarks' textual keywords. These also tell us their synonyms and geographic hierarchy. The ability to assign structure to tags makes tagging systems more useful. In addition, we propose an efficient indexing method for content-based landmark search. With all these, we demonstrate a set of interesting applications related to landmarks. In the future, we plan to develop more interesting applications using the discovered landmark ontology and make the visual search for landmarks more efficient.

References

1. Naaman, M., Harada, S., Wang, Q., Garcia-Molina, H., Paepcke, A.: Context data in geo-referenced digital photo collections. In: Proceedings of ACM Multimedia. (2004) 196–203
2. Kennedy, L.S., Naaman, M.: Generating diverse and representative image search results for landmarks. In: Proceedings of WWW. (2008) 297–306
3. Ahern, S., Naaman, M., Nair, R., Yang, J.: World explorer: Visualizing aggregate data from unstructured text in geo-referenced collections. In: Proceedings of ACM/IEEE JCDL. (2007)
4. Toyama, K., Logan, R., Roseway, A., Anandan, P.: Geographic location tags on digital images. In: Proceedings of ACM Multimedia. (2003) 156–166
5. Jaffe, A., Naaman, M., Tassa, T., Davis, M.: Generating summaries and visualization for large collections of geo-referenced photographs. In: Proceedings of MIR. (2006) 89–98
6. Rattenbury, T., Good, N., Naaman, M.: Towards extracting Flickr tag semantics. In: Proceedings of WWW. (2007) 1287–1288
7. Rattenbury, T., Good, N., Naaman, M.: Towards automatic extraction of event and place semantics from Flickr tags. In: Proceedings of ACM SIGIR. (2007) 103–110
8. Kennedy, L., Naaman, M., Ahern, S., Nair, R., Rattenbury, T.: How Flickr helps us make sense of the world: Context and content in community-contributed media collections. In: Proceedings of ACM Multimedia. (2007) 631–640
9. Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total recall: Automatic query expansion with a generative feature model for object retrieval. In: Proceedings of IEEE ICCV. (2007)
10. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: Proceedings of IEEE CVPR. (2007)
11. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: Proceedings of CVPR. (2008)
12. Hays, J., Efros, A.: IM2GPS: estimating geographic information from a single image. In: Proceedings of IEEE CVPR. (2008)
13. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Internatioanl Journal of Computer Vision **60**(2) (2004) 91–110
14. Sivic, J., Zisserman, A.: Video Google: A text retrieval approach to object matching in videos. In: Proceedings of IEEE ICCV. Volume 2. (2003) 1470–1477
15. Nistér, D., Stewénius, H.: Scalable recognition with a vocabulary tree. In: Proceedings of IEEE CVPR. Volume 2. (2006) 2161–2168
16. Schindler, G., Brown, M., Szeliski, R.: City-scale location recognition. In: Proceedings of IEEE CVPR. (2007)