Tour the World: building a web-scale landmark recognition engine

Yan-Tao Zheng[1], Ming Zhao[2], Yang Song[2], Hartwig Adam[2]
Ulrich Buddemeier[2], Alessandro Bissacco[2], Fernando Brucher[2]
Tat-Seng Chua[1], and Hartmut Neven[2]

[1] NUS Graduate Sch. for Integrative Sciences and Engineering, National University of Singapore, Singapore

[2] Google Inc., U.S.A

{yantaozheng, chuats}@comp.nus.edu.sg

{mingzhao, yangsong, hadam, ubuddemeier, bissacco, fbrucher, neven}@google.com

Abstract

Modeling and recognizing landmarks at world-scale is a useful yet challenging task. There exists no readily available list of worldwide landmarks. Obtaining reliable visual models for each landmark can also pose problems, and efficiency is another challenge for such a large scale system. This paper leverages the vast amount of multimedia data on the web, the availability of an Internet image search engine, and advances in object recognition and clustering techniques, to address these issues. First, a comprehensive list of landmarks is mined from two sources: (1) ∼20 million GPS-tagged photos and (2) online tour guide web pages. Candidate images for each landmark are then obtained from photo sharing websites or by querying an image search engine. Second, landmark visual models are built by pruning candidate images using efficient image matching and unsupervised clustering techniques. Finally, the landmarks and their visual models are validated by checking authorship of their member images. The resulting landmark recognition engine incorporates 5312 landmarks from 1259 cities in 144 countries. The experiments demonstrate that the engine can deliver satisfactory recognition performance with high efficiency.

1. Introduction

The touristic landmarks are easily recognizable and well-known sites and buildings, such as a monument, church, etc, as shown in Figure 1. They are the pivotal part of people's tours, due to their notable physical, cultural and historical features. The explosion of personal digital photography, together with Internet, has led to the phenomenal growth of landmark photo sharing in many websites like Picasa Web Album (picasa.google.com). With the vast amount of landmark images in the Internet, the time has come for computer vision to think about landmarks globally, namely to build a landmark recognition engine, on the scale of the entire globe. This engine is not only to visually recognize the presence of certain landmarks in an image, but also contributes to a worldwide landmark database that organizes and indexes landmarks, in terms of geographical locations, popularities, cultural values and social functions, etc. Such an earth-scale landmark recognition engine is tremendously useful for many vision and multimedia applications. First, by capturing the visual characteristics of landmarks, the engine can provide clean landmark images for building virtual tourism [14] of a large number of landmarks. Second, by recognizing landmarks, the engine can facilitate both content understanding and geo-location detection of images and videos. Third, by geographically organizing landmarks, the engine can facilitate an intuitive geographic exploration and navigation of landmarks in a local area, so as to provide tour guide recommendation and visualization.

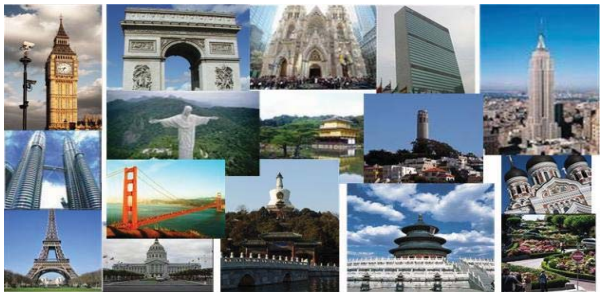


Figure 1. Examples of landmarks in the world.

To build such an earth-scale landmark recognition engine, the following issues, however, must be tackled: (a) there is no readily available list of landmarks in the world; (b) even if there were such a list, it is still challenging to col-

lect true landmark images; and (c) efficiency is a nontrivial challenge for such a large-scale system.

•**Discovering landmarks in the world:** It is not challenging to list a small number of most famous landmarks in the world. However, what is demanded here is a comprehensive and well-organized list of landmarks, across the entire planet. To achieve this goal, we explore two sources on the Internet: (1) the geographically calibrated images in photo sharing websites like picasa.google.com and panoramio.com; and (2) travel guide articles from websites, such as wikitravel.com. The first source contains a vast amount of GPS-tagged photos, together with their text tags, providing rich information about interesting touristic sites. Intuitively, if a large number of visually similar photos are densely concentrated on a geographical site, this site has a high probability to be a touristic landmark. The corresponding landmark names can then be mined from the geographic text tags of these images. Meanwhile, the popularity of these landmarks can be estimated by analyzing the number of uploaded photos, users and uploading time span, etc.

The landmark mining from the first source provides only a partial list, from the viewpoint of photo uploaders who have visited the landmarks and taken photos there. To complement the landmark list, we also exploit the second source of landmark information from travel guide articles in websites like wikitravel.com. The travel guide articles are authored and edited collaboratively by worldwide volunteers. The landmark list mining can be formulated as a task of text-based named entity extraction from the tour guide corpus. By exploiting these two sources of information, we can mine a more comprehensive list of landmarks. This is so because landmark is a perceptional and cognitive concept, which people of different background tend to perceive differently. Our experiments confirm this premise, by showing that the landmarks mined from GPS-tagged photos and travel guide articles have small overlap and complement each other.

•**Mining true landmark images:** While discovering the above list of landmarks, we also downloaded ∼21.4 million potential landmark images from two sources: (1) photo sharing websites, like picasa.google.com and panoramio.com and (2) Google Image Search. The challenge now is how to mine true landmark images out of the fairly noisy image pools. Our proposed approach relies on analyzing the visual similarity distribution among images. The premise is simple: the true images of a landmark tend to be visually similar. Thanks to the advanced object recognition techniques [10] [13], the image matching can handle variations in image capturing conditions, illuminations, scale, translation, clutter, occlusion and affine transformation in part. Our approach is to perform visual clustering

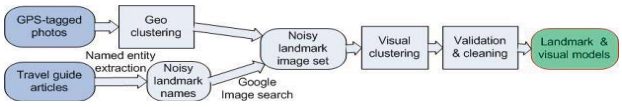


Figure 2. Overall framework.

[1] on the noisy image set. The resulting dense clusters of images are highly probable to be true landmark photos that depict the landmarks from similar perspectives. To further validate the resulting landmarks and visual clusters, we examine the authorship of images in the cluster. Namely, the images of a true landmark should come from different authors (uploaders or hosting webs) to reflect the popular appeal of the landmark.

•**Efficiency:** The whole pipeline in our system involves tremendous amount of images (∼21.4 million in our experiments). The resulting landmark recognition engine also incorporates a large number of landmarks and model images. Efficiency, therefore, becomes critical for both landmark model generation and landmark recognition of query image. Here, we accomplish efficiency by three means: (1) parallel computing of landmark models on multiple machines; (2) efficient clustering algorithm; and (3) efficient image matching by k-d tree indexing [1].

2. Overview and Preliminaries

As shown in Figure 2, the system processes two types of data sources: (1) a set of GPS-tagged photos $\mathbb{P} = \{p\}$; and (2) a corpus of travel guide articles $\mathbb{D} = \{d\}$. For the first source, a photo p is a tuple $(\theta_p, \wp_p, t_p, u_p)$, containing the unique photo ID θ_p, tagged GPS coordinates $\wp_p$ in terms of latitude and longitude, text tag t_p and uploader id u_p. The system performs clustering on photos' GPS $\wp_p$ to obtain the dense geo-clusters. The photos in one geo-cluster form a noisy image set $\mathcal{I}_1$, which probably contains images of one or several adjacent landmarks. The visual clustering is then performed on the noisy image set $\mathcal{I}_1$. The resulting cluster is deemed to contain true images of a landmark, if it passes the validation on its constituent image authorship. For each visual cluster of GPS-tagged photos, the corresponding landmark name can be mined by analyzing the constituent photos' geographic text labels.

The second data source is the travel guide corpus $\mathbb{D} = \{d\}$, where $d = \{e_d(i,j), t_d(i,j)\}$ is a semi-structured HTML document with a structure tree e_d, derived from the hierarchy of HTML tags and associated attributes [16]. $e_d(i,j)$ is the j^{th} node at level i of e_d and $t_d(i,j)$ is the text terms of node $e_d(i,j)$. For the travel guide corpus $\mathbb{D}$, the system performs named entity extraction, based on the semantic clues embedded in the document structure to extract a noisy list of landmark candidates. The text associated with each landmark candidate is then used as query for

[1] Visual clustering means clustering using image visual features.

Google Image Search to generate a noisy image set $\mathcal{I}_2$. The true landmark images are then mined by performing visual clustering on $\mathcal{I}_2$.

The final step is to clean the visual clusters, by training a photographic v.s. non-photographic image classifier and a multi-view face detector. The images that are detected as non-photographic or with a overly large area of human face are deemed to be outliers.

To obtain the GPS coordinates, the landmark is fed into the geo-coding service of Google Maps.

3. Discovering Landmarks in the World

Here, we formulate the worldwide landmark mining as a large-scale multi-source and multi-modal data mining on the vast amount of noisy tourism related multimedia data on the Internet. Specifically, we explore two sources of information: (1) GPS-tagged photos from photo-sharing website picasa.google.com and panoramio.com; and (2) travel guide articles from wikitravel.com.

3.1. Learning landmarks from GPS-tagged photos

Our premise here is that the true landmark should correspond to a set of photos that are geographically adjacent, visually similar and uploaded by different users. Hence, our approach is to first cluster photos geographically and then perform visual clustering on the noisy image sets of geo-clusters to discover landmarks.

• **Geo-clustering:** We perform the agglomerative hierarchical clustering on the photos' GPS coordinates $\wp$. The inter-cluster distance is defined as the distance between the cluster centers, which is the average of its images' GPS coordinates. Each geo-cluster then goes through a validation stage to ensure that it is reasonably probable to include a touristic landmark. The validation criterion is that the unique number of authors or uploaders u_p of photos p in the geo-cluster is larger than a pre-determined threshold. This validation criterion can filter out photos of buildings and sites that have little popular appeal. For example, an enthusiastic homeowner may post many pictures of his newly built house that has no popular attraction. The geo-cluster of his house is unlikely to be substantial, when compared to the popular landmarks whose photos are posted by many users of photo-sharing websites.

• **Learning landmark names from visual clusters:** For the noisy image set $\mathcal{I}_1$ of each geo-cluster, we then perform visual clustering, which will be introduced in Section 4 in detail. After visual clustering, we extract text tags t_p of each photo p in the visual cluster by filtering stop words and phrases. We then compute the frequency of n-grams of all text tags in each visual cluster. The resulting n-grams with the highest frequency is regarded as the landmark name for the visual cluster. The rationale here is that photo uploaders are willing to spend effort on tagging their own tour photos with landmark names. The photos are rarely noise, when they are visually similar, geographically adjacent and sharing the same text tags at the same time.

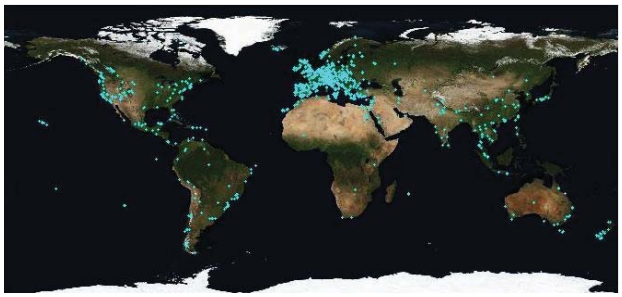


Figure 3. The distribution of landmarks mined from GPS-tagged photos in picasa.google.com and panoramio.com.

• **Observation:** The GPS-tagged photos yield ∼140k geo-clusters and ∼14k visual clusters with text tags, from which 2240 landmarks from 812 cities in 104 countries are mined. Figure 3 displays the distribution of these landmarks. As shown, most landmarks are located in Europe and North America. We attribute this distribution bias to the user community of picasa.google.com and panoramio.com, as most users are located in Europe and North America.

3.2. Learning landmarks from travel guide articles

Before mining landmarks from tour guide corpus on the Internet, we define the geographical hierarchy for landmarks, as the tours and landmarks are, in essence, about geography. Here, we assume the following geographical hierarchy:

• $landmark \in city \in country \in continent$

This hierarchy makes city as the unit containing landmarks. The concept of "city" here is flexible. It does not only indicate urban area but also larger metropolitan areas with suburbs and satellite cities, which is consistent with its definition used in wikitravel.com. With the hierarchy, we can then recursively extract city names from countries in six continents on the earth (except Antarctica). The travel guide articles of these cities can then be downloaded from wikitravel.com accordingly.

The task now is reduced to extract landmark names from the city tour guide corpus $\mathbb{D} = \{d\}$, where $d = \{e_d(i,j), t_d(i,j)\}$ is a city tour guide HTML file. The interior nodes of the structure tree e_d correspond to tag elements of documents and the leaf nodes store the text [16]. Landmark name extraction is equivalent to classifying the text $t_d(i_{leaf}, j)$ of leaf nodes $e_d(i_{leaf}, j)$ to be either landmark or non-landmark names. Here, we utilize a simple but effective landmark classifier, based on a set of heuristic rules. For each leaf $e_d(i_{leaf}, j)$ and its text $t_d(i_{leaf}, j)$, if they satisfy all the following criteria, then text t_d is deemed to be a landmark candidate.

1. $e_d(i_{leaf}, j)$ is within the Section "See" or "To See" in the tour guide article.

2. $e_d(i_{leaf}, j)$ is the child of a node indicating "bullet list" format, as landmarks tend to be in a bullet list.

3. $e_d(i_{leaf}, j)$ indicates the bold format, as the landmark name is usually emphasized in bold.

The mined landmark name, together with its city, is then used as query for Google Image Search to retrieve a set of potential landmark images $\mathcal{I}_2$. The true landmark images are learned from $\mathcal{I}_2$.

• **Observation:** By utilizing the geographical hierarchy in wikitravel.com, we extract 7315 landmark candidates from 787 cities in 145 countries and 3246 of them can be associated with valid visual clusters. Figure 4 displays the distribution of these landmarks. As shown, the landmarks are more evenly distributed across the world than the ones mined from GPS-tagged photos. This is so because the community of wikitravel.com is more diverse. Most photos were uploaded by tourists that took them, while the tour guide article can be authored or edited by anyone who has the knowledge about the touristic site.

3.3. Validating landmarks

The resulting landmark candidates can be noisy. Two validations are performed to ensure its correctness. First, the landmark candidate is filtered for error-checking, if it is too long or most of its words are not capitalized. This is so because a true landmark name should not be too long and most of its words are generally capitalized. The second validation on landmark is to check its associated visual clusters. The validation criterion is the number of unique authors (uploaders or hosting webpages) of images in the cluster, which reflects the popular appeal of landmarks. Similar to the validation of geo-clusters, the number of unique photo authors must be above a pre-determined threshold. Otherwise, the visual clusters and their associated landmark candidates will be deemed false.

4. Unsupervised Learning of Landmark Images

Given the noisy sets $\mathcal{I} = \mathcal{I}_1 \cup \mathcal{I}_2$ of potential landmark images from geo-clusters and Google Image Search, our task now is to learn true landmark images from noisy image pools. This not only serves to construct visual models of landmarks, but also facilitates landmark discovery and validation process.

To mine true landmark images in the noisy image set $\mathcal{I}$, our approach relies on analyzing the visual similarity distribution among images. The rationale is that each true landmark photo, in essence, represents a view of landmark from certain perspective and capturing conditions. Due to the geometric closeness, these photos will naturally form view clusters. The true landmark images can, therefore, be discovered by performing clustering on image set $\mathcal{I}$, and the resulting clusters are reasonably probable to contain true landmark images. Prior to presenting our clustering technique, we introduce the object matching method we use first.

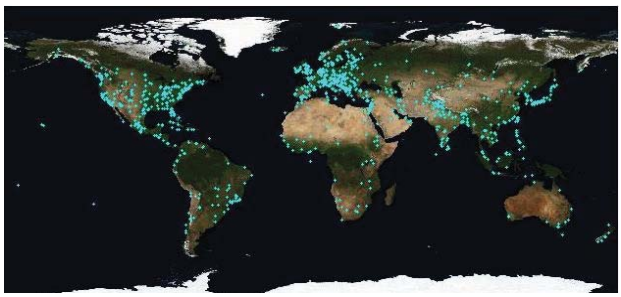


Figure 4. The distribution of landmarks extracted from tour guide corpus in wikitravel.com.

4.1. Object matching based on local features

Given two images I_α and I_β, we match them by comparing their local features. The local feature consists of 2 parts: interest points and their descriptors. Here, we exploit the Laplacian-of-Gaussian (LoG) filters [11] to detect interest points. For local descriptor, we utilize an approach similar to SIFT [9], by computing a 118 dimension Gabor wavelet texture features on the local region. A Principle Component Analysis (PCA) [2] is then performed to reduce the feature dimensionality to 40, for efficiency purpose. The match interest points of two images are then verified geometrically by an affine transformation [9]. The matching outputs are the match score and match region, which is defined by the interest points contributing to the match.

The match score is estimated by $1 - P_{FP\alpha\beta}$, where $P_{FP\alpha\beta}$ is the probability that the match between I_α and I_β is a false positive. $P_{FP\alpha\beta}$ is computed by using the probabilistic model in [10]. First, a probability p is assumed to be the chance of accidentally matching two local features from I_α and I_β. The probability $P_{FP}(\text{feature matches} \,|\mathrm{I}_\alpha \neq \mathrm{I}_\beta)$ of at least m accidental feature matches out of n features in the match region can then be estimated by using a cumulative binomial distribution, as below:

$$P_{FP}(\text{feature matches}|\mathrm{I}_\alpha \neq \mathrm{I}_\beta) = \sum_{j=m}^{n} \binom{n}{j} p^j (1-p)^{n-j} \tag{1}$$

$P_{FP\alpha\beta} \doteq P_{FP}(\mathrm{I}_\alpha \neq \mathrm{I}_\beta|\text{feature matches})$ can then be estimated by Bayes Theorem [2].

4.2. Constructing match region graph

After performing object matching on all images in the set, we obtain an undirected weighted match region graph,

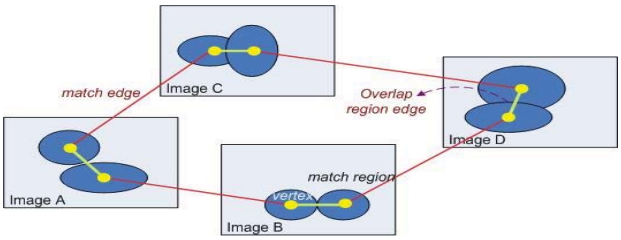


Figure 5. Undirected weighted match region graph.

in which the vertexes are match regions, as shown in Figure 5. The edges connecting regions are classified into two types: match edge and region overlap edge. Match edge connects match regions of two different images, while region overlap edge connects regions in the same image.

• **Estimating edge weight:** The edge weight is quantified by its length. For match edge of region i and j, its length d_{ij} is defined as below:

$$d_{ij} = -\frac{1}{log(P_{FPij})} \tag{2}$$

where P_{FPij} is the probability that the match between region i and j is a false positive, as introduced in Section 4.1.

The length of region overlap edge is determined by the spatial overlap of two regions, which is specifically defined as below.

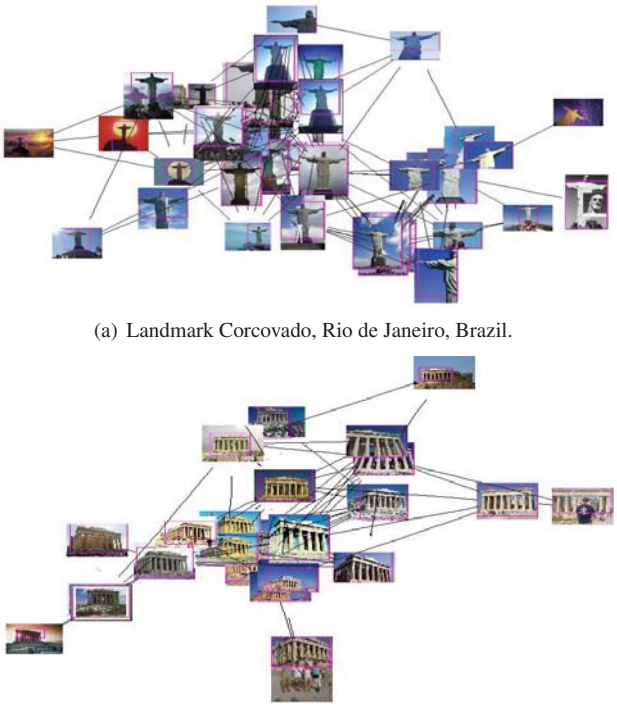
$$d_{ij} = f_d \frac{|\underline{r}_i - \underline{r}_j|_{L2}}{\sqrt{s_i + s_j}} \tag{3}$$

where $\underline{r}_i = \frac{1}{K}\sum_{k=1}^{K} r_{ik}$, the center of gravity of region i, $s_i = \frac{1}{K}\sum_{k=1}^{K}(|r_{ik}|_{L2}^2 + 2\sigma_s^2 s_{ik}^2) - |\underline{r}_i|_{L2}^2$, the squared expansion of region i, (r_{ik}, s_{ik}) are the location and scale of interest points comprising region i and K is the number of feature matches. f_d is a factor to adjust the two different distance measures for match and region overlap edges. σ_s is a scale multiple to account for the size of the image patch used to compute the descriptor relative to the interest point scale s_{ik}.

4.3. Graph clustering on match regions

As the distance between any two match regions in the image set has been established, the clustering on the undirected weighted region graph can then be performed to discover regions of same or similar landmark views. Since we do not have a priori knowledge of the number of clusters, the k-means [2] like clustering techniques are unsuitable. We, therefore, exploit the hierarchical agglomerative clustering [2]. For efficiency purpose, we utilize the single linkage inter-cluster distance to define the distance of region C_n and C_m as $d(C_n, C_m) = \min_{i \in C_n, j \in C_m} d_{ij}$.

Figure 6 displays the cluster examples of "Corcovado" and "Acropolis". As shown, one byproduct of clustering is the canonical views of landmarks. If a photo has dense



(a) Landmark Corcovado, Rio de Janeiro, Brazil.

(b) Landmark Acropolis, Athens, Greece.

Figure 6. Examples of region graph cluster.

connections with other photos, then the view in this photo tends to be canonical and the photo can be selected as an iconic photo for the landmark.

4.4. Cleaning visual model

Our observation also shows that one major visual cluster outlier are map images. This is so because landmark is a geographic concept too. When searching landmarks in Google Image Search, the maps of its geographic district or city are also likely to be returned, as shown in Figure 8. To prune these outliers, we exploit a photographic v.s. non-photographic image classifier. The classifier is trained based on Adaboost algorithm over low level visual features of color histogram and hough transform. Moreover, we also adopt a multi-view face detector [15] to filter out photos with overly large area of face. This is to ensure the purity of landmark models, by pruning photos dominated by people in front of landmarks.

5. Efficiency Issues

In the processing pipeline, the geo-clustering of GPS-tagged photos and landmark mining from tour guide corpus do not demand high efficiency requirement, due to the low dimensionality of GPS coordinates and relatively small tour guide corpus size. However, the large amount (∼21.4 million) of raw input images and large magnitude of land-

Figure 8. Map outlier cluster of "Mayapan, Mérida, Mexico".

mark models make efficiency essential in two aspects: (1) the landmark image mining and (2) landmark recognition of query images. To achieve efficiency, we exploit the following three measures.

• **Parallel computing to mine true landmark images:** The visual clustering process on each noisy image set $\mathcal{I}$ does not interfere with each other. This enables us to speed up the clustering process drastically by running parallel visual clustering on multiple machines.

• **Efficiency in hierarchical clustering:** By adopting single linkage, the shortest path between two clusters is equal to the shortest path of two regions in clusters, which has been computed in the phase of image matching. The clustering process is then equivalent to erasing graph edges above a certain distance threshold and collecting the remaining connected region sets as clusters.

• **Indexing local feature for matching:** To achieve fast image matching, we adopt the k-d tree [1] to index local features of images. This allows the local feature matching time to become sub-linear, thus enabling efficient recognition of query images. In our experiments, the time it takes to recognize landmark in a query images is only ∼0.2 seconds in a P4 computer.

6. Experiments and Discussion

We employ ∼20 million GPS-tagged photos from picasa.google.com and panoramio.com to construct noisy image set $\mathcal{I}_1$ for each geo-cluster. We also query the landmark candidates mined from tour guide corpus in Google Image Search to construct noisy image set $\mathcal{I}_2$ from first 200 returned images. The total number of images amounts to ∼21.4 million. The object matching and graph clustering are performed on each image set $\mathcal{I} = \mathcal{I}_1 \cup \mathcal{I}_2$ to mine true landmark images and construct landmark visual models.

We evaluate the resulting landmark recognition engine in three aspects: (1) the scale and distribution of mined landmarks; (2) the efficacy of clustering for landmark image mining, namely the correctness of landmark visual clusters; and (3) the landmark recognition accuracy on query images.

6.1. Statistics of mined landmarks

The mining on GPS-tagged photos delivers 2240 validated landmarks, from 812 cities in 104 countries. The tour guide corpus yields 3246 validated landmarks, from 626 cities in 130 countries. Our initial conjecture was that these two lists should be similar. However, after careful comparison, only 174 landmarks are found to be common in both

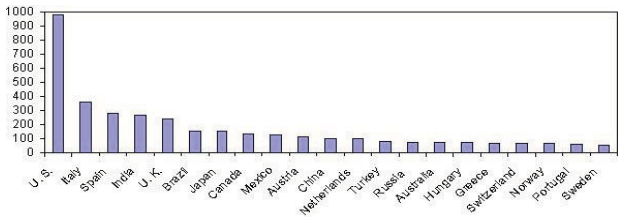


Figure 9. Top 20 countries with the largest number of landmarks.

lists. This finding is surprising but rational. This is because the landmark is a perceptional and cognitive concept, in which different communities of people perceive landmarks differently. The landmarks mined from GPS-tagged photos reflect the perception of tourists who have visited the touristic site and taken photos there. On the other hand, the landmarks mined from online tour guide corpus indicates the perception of web authors or editors, who may not necessarily visit the landmarks, but have some knowledge of them. The 174 landmarks common in two lists are most famous ones, like Eiffel Tower and Arc de Triomphe, etc.

The combined list of landmarks consists of 5312 unique landmarks from 1259 cities in 144 countries. The landmark distribution is shown in Figure 7. As shown, the discovered landmarks are more densely distributed in North America and Europe than in South America, Asia and Africa. This is attributed to the fact that our processing language focuses on English only. Consequently, the resulting landmarks tend to be those popular among the English speakers only. Figure 9 displays the top 20 countries with the largest number of landmarks. Among the 20 countries, United States has 978 landmarks, which is absolutely higher than the rest. This is attributed to its large geographical area and enormous tourism sites, and more importantly, its high Internet penetration rate and large Internet user base. Nevertheless, the landmarks are the results of mining multimedia data on the Internet. Another interesting observation is that the number of landmarks in China amounts to 101 only, which is clearly under-counted. This also manifests that building a world-scale landmark recognition engine is not only a computer vision task, but also a multi-lingual data mining task.

6.2. Evaluation of landmark image mining

The landmark image mining is achieved by the visual clustering algorithms described in Section 4. Here, we set the minimum cluster size to 4. The visual clustering yields ∼14k visual clusters with ∼800k images for landmarks mined from GPS-tagged photos and ∼12k clusters with ∼110k images for landmarks mined from tour guide corpus. Figure 6 some visual cluster examples. More visual cluster examples are illustrated in the supplementary material.

To quantify the clustering performance, 1000 visual clusters are randomly selected to evaluate the correctness and

Figure 7. Distribution of landmarks in recognition engine.

Figure 10. Examples of positive landmark testing images.

purity of landmark visual models. Among the 1000 clusters, 68 of them are found to be negative outliers, most of which are landmark related maps, logos and human profile photos. We then perform the cluster cleaning, based on a photographic v.s. non-photographic image classifier and a multi-view face detector. The classifier is trained based on $\sim$5000 photographic and non-photographic images, while the face detector is based on [15]. After cleaning, the outlier cluster rate drops from 0.68% (68 out of 1000) to 0.37% (37 out of 1000).

6.3. Evaluation of landmark recognition

Next, we evaluate the performance of landmark recognition on a set of positive and negative query images.

• **Experimental setup:** The positive testing image set consists of 728 images from 124 randomly selected landmarks. They are manually annotated from images that range from 201 to 300 in the Google Image Search result and do not host in picasa.google.com or panoramio.com. This testing image set is considered challenging, as most landmark images are with large variations in illumination, scale and clutter, as shown in Figure 10. For the negative testing set, we utilize the public image corpus Caltech-256 [4] (without "eiffel tower", "golden gate bridge", "pyramid" and "tower pisa" categories) and Pascal VOC 07 [3]. Together, the negative testing set consists of 30524 (Caltech-256) + 9986 (Pascal VOC 07) = 40510 images in total.

(a) Landmarks can be locally visually similar

(b) Regions, like U.S. flag, in landmark model can be non-representative.

(c) Negative images and landmark model images can be similar.

Figure 11. False Landmark matches.

The recognition is done by local feature matching of query image against model images, based on the nearest neighbor (NN) principle. The match score is measured by the edge weight between query image and its NNs in the match region graph. A match is found, when the match score is larger than the threshold $d_{thres} = 5$.

• **Recognition accuracy:** For the positive testing image set, 417 images are detected by the system to be landmarks, of which 337 are correctly identified. The accuracy of identification is 80.8%, which is fairly satisfactory, considering the large number of landmark models in the system. This high accuracy enables our system to provide landmark recognition to other applications, like image content analysis and geo-location detection. The identification rate (correctly identified / positive testing images) is 46.3% (337/728), which is regarded to be moderately satisfactory, considering the fact that the testing images are with large visual variations in scale, illumination and clutter, etc. We at-

tribute the false landmark identification to the fact that some landmarks have similar local appearance, as shown in Figure 11 (a). This local appearance similarity leads to false match among landmark images.

For the negative testing set, 463 out of 40510 images are identified with some landmarks and the false acceptance rate is only 1.1%. After careful examination, we find that most false matches occur in two scenarios: (1) the match is technically correct, but the match region is not representative to the landmark; and (2) the match is technically false, due to the visual similarity between negative images and landmark. The first scenario is illustrated in Figure 11 (b), in which the U.S. flag image is matched with New York Stock Exchange. This is, in fact, a problem of model generation. Namely, the inclusion of U.S. flag in the landmark model leads to the false match. The second scenario is illustrated in 11 (c), in which the chess board image is matched to "Museo dell'Automobile, Itaty", due to their visual similarity. This is actually is a problem of image feature and matching mechanism. Ideally, a more distinctive feature and matching mechanism are demanded.

7. Related Work

The touristic landmarks have interested many computer vision researchers. Snavely et al. [14] and Goesele et al. [5] employed the geometric constraints to construct 3D visualization of landmarks, based on a set of relatively clean landmark photos. Our landmark recognition engine, in fact, can provide input data to these 3D reconstruction systems and enables them to be scalable to a large number of landmarks. To mine a clean set of landmark images, Li et al. [8], Quack et al. [12] and Kennedy and Naaman [7] employed the community photo collections, by analyzing the geometric, visual, geographical (GPS tags) and textual cues. Contrasting to [8], [12] and [7], the principal focus of our approach is to explore landmarks at a world-scale. To the best of our knowledge, this is the first approach to model and recognize landmarks in the scale of the entire planet Earth. In this aspect, we share similar vision with Hays and Efros [6], which estimated the geographic information from an image at world scale. Our focus, however, is to capture the visual characteristics of worldwide landmarks, so as to facilitate landmark recognition, modeling, 3D reconstruction, and furthermore image and video content analysis.

8. Conclusion and Future Work

The phenomenal emergence of tourism related multimedia data in the Internet, such as the GPS-tagged photos and tour guide web pages, has prompted computer vision researchers to think about landmarks globally. Here, we build a world-scale landmark recognition engine, which organizes, models and recognizes the landmarks on the scale of the entire planet Earth. Constructing such an engine is, in essence, a multi-source and multi-modal data mining task. We have employed the GPS-tagged photos and online tour guide corpus to generate a worldwide landmark list. We then utilize $\sim$21.4 M images to build up landmark visual models, in an unsupervised fashion. The landmark recognition engine incorporates 5312 landmarks from 1259 cities in 144 countries. The experiments demonstrate that the engine can deliver satisfactory recognition performance, with high efficiency.

One important issue remains open. The multi-lingual aspect of landmark engine is neglected. Here, our processing language is English only. The multi-lingual processing can help to discover more landmarks and collect more clean landmark images, as many landmarks are more widely broadcasted in their native languages in the Internet.

References

[1] J. L. Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, September 1975. 2, 6

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, August 2006. 4, 5

[3] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC. 7

[4] A. H. G. Griffin and P. Perona. Caltech-256 object category dataset. Technical report, California Institute of Technology, 2007. 7

[5] M. Goesele, N. Snavely, B. Curless, H. Hoppe, and S. M. Seitz. Multi-view stereo for community photo collections. In *Proc. of IEEE Conf. on Computer Vision*, 2007. 8

[6] J. Hays and A. Efros. im2gps: estimating geographic information from a single image. In *Proc. of Conf. on Computer Vision and Pattern Recognition*, 2008. 8

[7] L. S. Kennedy and M. Naaman. Generating diverse and representative image search results for landmarks. In *Proc. of Conf. on World Wide Web*, pages 297–306, Beijing, China, 2008. 8

[8] X. Li, C. Wu, C. Zach, S. Lazebnik, and J.-M. Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *ECCV (1)*, pages 427–440, 2008. 8

[9] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 20, pages 91–110, 2003. 4

[10] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV 99*, pages 1150–1157, 1999. 2, 4

[11] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004. 4

[12] T. Quack, B. Leibe, and L. V. Gool. World-scale mining of objects and events from community photo collections. In *Proc. of Conf. on Content-based Image and Video Retrieval*, pages 47–56, New York, NY, USA, 2008. 8

[13] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of ICCV*, page 1470, 2003. 2

[14] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM Transactions on Graphics*, pages 835–846. Press, 2006. 1, 8

[15] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of Conf. on Computer Vision and Pattern Recognition*, volume 1, pages I–511–I–518 vol.1, 2001. 5, 7

[16] J. Yi and N. Sundaresan. A classifier for semi-structured documents. In *Proc. of Conf. on Knowledge Discovery and Data Mining*, pages 340–344, New York, NY, USA, 2000. 2, 3