GPS Estimation for Places of Interest From Social Users' Uploaded Photos

Jing Li, Xueming Qian, *Member, IEEE*, Yuan Yan Tang, *Fellow, IEEE*, Linjun Yang, and Tao Mei, *Senior Member, IEEE*

***Abstract*—Social media has become a very popular way for people to share their photos with friends. Because most of the social images are attached with GPS (geo-tags), a photo's GPS information can be estimated with the help of the large geo-tagged image set while using a visual searching based approach. This paper proposes an unsupervised image GPS location estimation approach with hierarchical global feature clustering and local feature refinement. It consists of two parts: an offline system and an online system. In the offline system, a hierarchical structure is constructed for a large-scale offline social image set with GPS information. Representative images are selected for each GPS location refined cluster, and an inverted file structure is proposed. In the online system, when given an input image, its GPS information can be estimated by hierarchical global clusters selection and local feature refinement in the online system. Both the computational cost and GPS estimation performance demonstrates the effectiveness of the proposed hierarchical structure and inverted file structure in our approach.**

***Index Terms*—BoW, GPS Estimation, Hierarchical Structure, Inverted File Structure, k-NN, Social Media, User.**

I. Introduction

With the development of communication technology, more and more digital devices, such as cameras and smart-phones, offer global positioning system (GPS) integration. Large quantities of images taken by users are shared on social media websites such as Facebook and Flickr every day. To make it more convenient to administrate resources of images, some additional information such as the times and GPS locations where they were taken should be provided, which leads to the problem of automatic GPS estimation in the web images.

Currently, the GPS information of social images has been widely used in many applications such as content browsing [1], [27], image annotation [6], [7], [37], image search [8]–[10], and localization [45]. Qian *et al.* have shown that using the GPS information of users' uploaded photos is helpful for improving users' vocabulary tagging performances [37]. Due to the advantages of GPS attached to images in both industrial and academic areas [10], Google and Flickr suggest that users geo-tag their shared images by dragging them onto the map. However, this kind of GPS assignment approach leads to large errors in GPS location. There are some situations in which find beautiful photos without any idea about the locations where they were taken. With the help of large-scale geo-tagged photos shared in social media, automatic image GPS location estimation can be achieved.

IM2GPS made the first attempt to estimate the GPS location for a given image by utilizing visual matching in a large geo-referenced image dataset [2]. Kalogerakis *et al.* further proposed methods of incorporating single image matching with sequential data to improve the estimation accuracy [28]. Zheng developed a worldwide landmark recognition system [29], which utilized a predefined landmark list to query online image search engines and select candidate images. Then they re-clustered and pruned the results to estimate the GPS information of the landmark [30], [31]. Moreover, the Placing Task makes use of attached information, such as tags and user descriptions to estimate the GPS of an image or video frame [32]–[34], [39]–[43].

Although a great deal of research effort has been devoted to image GPS location estimation, the task is still very challenging, especially when we only have the image without any supplementary information. What we can resort to is utilizing its visual information to perform GPS estimation for an input image. Both image GPS estimation performance and computational cost should be considered and meet the requirements of real-time applications. To get satisfactory image GPS estimation performance, local feature matching is required. However, the local feature matching between the input image and all the geo-tagged images in the database is extremely computational intensive. It is likely that images taken at different locations have similar appearance, but they have different local features. For example, images of churches taken at different places have similar color or texture patterns. Thus, to estimate the GPS of an image that contains a church, we first find groups of images containing churches, and then determine which photos have iden-

Manuscript received February 06, 2013; revised April 29, 2013; accepted May 02, 2013. Date of publication September 04, 2013; date of current version November 13, 2013. This work was partly performed when X. Qian was visiting at Macau University, China. This work was supported in part by National Natural Science Foundation of China (NSFC) Project No.60903121, No.61173109, No.61273244, Microsoft Research Asia, the Multi-Year Research Grants of University of Macau MYRG205(Y1-L4)-FST11-TYY and No. MYRG187(Y1-L3)-FST11-TYY and Start-up Research Grant of University of Macau SRG010-FST11-TYY, as well as the Science and Technology Development Fund (FDCT) of Macau FDCT-100-2012-A3. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Chong-Wah Ngo.

J. Li and X. Qian (corresponding author) are with the SMILES LAB at the School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China (e-mail: lijing.1@stu.xjtu.edu.cn; qianxm@mail.xjtu.edu.cn).

Y. Y. Tang is with FST of Macau University, Macau, China (e-mail: yytang@umac.edu.cn).

L. Yang and T. Mei are with Microsoft Research Asia, Beijing, China (e-mail: linjuny@microsoft.com; tmei@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2013.2280127

tical churches with the input image, and finally use the GPS information of the identical church images to estimate the location of the input image.

In this paper, we propose a hierarchical algorithm for estimating the GPS location of an image by using a purely unsupervised data-driven approach. First, we classify the input image into several candidate clusters with similar color or texture patterns. Then the input image is further attributed to a set of GPS location refined clusters. Finally, we use local feature matching to determine its accurate GPS location. The main contributions of this paper are as follows: 1) Building a hierarchical structure for a geo-tagged dataset by using both visual features and GPS information. 2) Proposing a hierarchical global feature classification and local feature refinement based GPS estimation approach. 3) Adopting the inverted file structure and selection of representative images for each GPS location to guarantee estimation speed and accuracy.

When compared with a preliminary version[35], we have made the following enhancements: 1) We have conducted a more comprehensive survey of related work; 2) we have proposed an inverted file structure for representative images to reduce computational cost; and 3) provided more experimental results and evaluation, including extending a large geo-tagged image set and utilizing cross validations. The rest of the paper is organized as follows: related work on image GPS estimation is reviewed in Section II. Section III introduces the system overview of the proposed image GPS estimation approach. Section IV examines the offline system for the geo-tagged image set. Section V looks at the online system for estimating the GPS location of an input image. Experiments and discussions are shown in Section VI. Conclusions are drawn in Section VII.

II. Related Work

IM2GPS, proposed by Hays and Efros, was a direct feature matching based approach for the GPS estimation of an input image [2]. In IM2GPS, the distances of an input image to all the geo-tagged images are measured in a low-level visual feature space. By ranking the distances in ascending order, the K-nearest neighbors (KNN) are selected to improve image GPS estimation accuracy. Then, mean-shift clustering complements the GPS locations of the images of the selected K-nearest neighbors. Finally, the cluster with the highest cardinality is selected and its GPS location is assigned to the input image [2].

Li *et al.* utilized multi-class SVM classifiers using bag-of-word features for large-scale image location estimation [26]. They also showed that by adding textural features such as tags, the performance can be improved. For an image without textual information, they had to use the sole visual feature for image GPS estimation. The computational costs are extremely high for the training of both model parameters. Also, when the dataset is extended, the models need to be trained again. Quack also proposed an approach for estimating the location of an image by utilizing the method of local feature matching [11]. The feature matching based GPS estimation approach is also very computational intensive when the scale of the dataset is very large. To speed up the estimation process, user interaction is required to confine the locations of the input images to a rough geographic area [11]. If the rough geographic area that the user assigned has a large error, then both the image GPS estimation performances and the computational cost results will be affected.

Actually, existing image classification and image retrieval approaches can be adopted to fulfill image GPS estimation [19], [20]. The main process can be as follows: first, find images similar to the input image, and then assign the GPS location of the visually similar images to the GPS location of the input image. From this point of view, an existing example-based image retrieval approach can be utilized in GPS location estimation for an input image. Zhang *et al.* proposed a spatial coding based image retrieval approach by building the contextual visual vocabulary [20]. By using inverted construction, the computational cost is low but produces a good performance. Techniques that generate 3-D models from large-scale geo-tagged photos are related to GPS location estimation [23]–[25]. Image retrieval is carried out by generating 3-D models and translating the query image into a 3-D pattern[23], [24]. Park el al. proposed a method of viewing direction determination by utilizing Google Street View and Google Earth satellite [25].

Placing Task (http://www.multimediaeval.org/) is a benchmark initiative devoted to the problem of placing multimedia that was first organized in 2009. Placing task invites participants to propose approaches to solve the problem of automatic annotation of video lacking geographical data [32]–[34]. In Placing Task 2012, Trevisiol *et al.* provided a method to identify the geographic location of videos by utilizing the attached tags[33]. They utilize the key frame of the video as a query to accomplish the retrieval and utilize the GPS information of the best match results as the GPS information of the key frame. However, their results show that when only utilizing image content that the estimation performance is not satisfactory. Laere proposed a two-step process for geo-referencing tagged resources [39]. They first use language models to find an area that is likely to contain the location of the resource. Then, the location is determined by choosing the most similar resources in the second step. In their method, tags are taken into consideration to measure the similarity between the input and offline images. Tzy carried out video geo-referencing by combining textual features and visual features [40]. Text processing, visual processing, and data/information fusion are the three steps for predicting an unseen query video. The visual processing module ranks the video in the training set by visual similarity with the test video. The textual processing module works in a similar fashion. The fusion module combines the results in both visual and textual processing using rank aggregation. Kelm *et al.* proposed a framework to geo-tag video using textual and visual information of shared media [41]. As for the textual information, they detect the language and translate the text into English using the web service Google Translate. Probabilistic latent semantic analysis (pLSA) and collaborative systems are utilized to process the textual information. A support vector machine (SVM) is trained to process the visual features, color, and edge features. Li proposed a pure image content-based approach for video geo-referencing [42]. They partition the world map into regions based on external data sources such as climate and biomes data. As for the visual content, they use the key frames of videos and represent each frame by its visual features. A support vector machine with

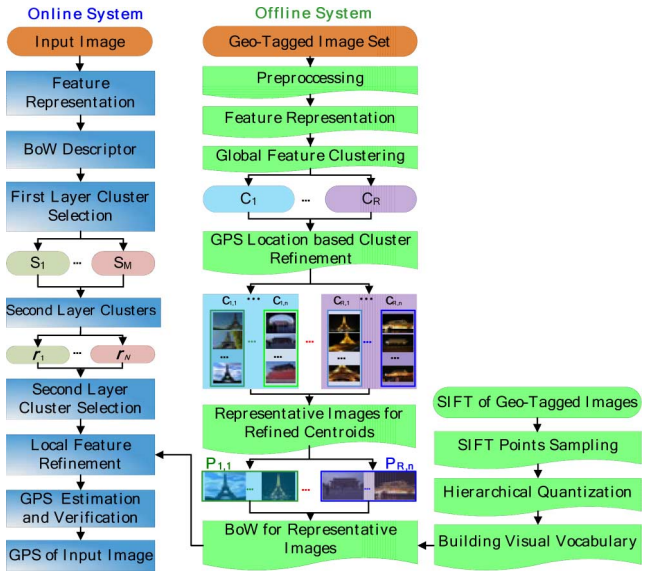


Fig. 1. Block diagram of the GPS estimation system. It consists of online and offline systems. The offline system aims at indexing the large scale geo-tagged image set. The online system is to estimate the GPS of input image.

RBF kernel is utilized to choose the most similar regions. Kelm *et al.* also addressed the problem of video geo-referencing [43]. They make use of external resources like gazetteers to extract homonyms in the metadata. Visual and textual features are used to identify similar content. The videos' locations are classified into possible regions by utilizing a method of fusing the visual and textual features. Flickr videos are tagged with the geo-information of the most similar training image within the regions that were previously filtered by the probabilistic model for test video.

III. System Overview

Intuitively, the GPS location of an input image can be obtained by comparing its visual content to large-scale, geo-tagged image sets. However, feature matching based image GPS estimation approaches are computationally intensive. To speed up the estimation process, we propose a fast GPS estimation algorithm that uses the hierarchical structure [35] and inverted file structure for the geo-tagged images. The introduced hierarchical structure converts 'selection of the most similar image from the geo-tagged image dataset' to 'selection of hierarchical clusters from the dataset'. As the number of clusters is much smaller than the number of geo-tagged images, this conversion is time saving. The block diagram of our approach is shown in Fig. 1. It consists of online and offline systems.

The offline system aims to index the large-scale geo-tagged image datasets. It consists of the following six parts: 1) preprocessing to remove noisy images, 2) feature representation by extracting global and local features, 3) clustering the images into R categories utilizing global features, 4) obtaining GPS location refined centroids (i.e., each centroid corresponds to an identical GPS location) for each first layer cluster, 5) selecting representative images for each refined centroid, and 6) building inverted files for the representative images in each refined centroid based on the BoW of the SIFT descriptor. The detailed steps of the offline system are presented in Section IV.

The online system estimates the GPS location of an input image. It carries out the following four steps after global and local feature extraction: 1) first layer cluster selection, 2) second layer centroid selection, 3) local feature refinement, and 4) GPS location estimation and verification. In the first step, the input image is assigned to one or more of a number of clusters. Images in the same cluster are visually similar. In the second step, the input image is classified into a number of GPS location refined centroids. In the third step, the local feature refinement can also help improve the accuracy of GPS location estimation. Also, as the local refinement is confined to a much smaller scale compared to the whole dataset, it is very time efficient. What is more, the local refinement can help determine whether the input image's location is contained offline or not, which makes our system more robust. The detailed steps of the online system are presented in Section V.

IV. The Offline System

A. Preprocessing for the Dataset

Some of the crawled geo-tagged images are too bright or too dark (such as several shining stars in a black sky) or too smooth (such as a pure blue sky). These kinds of images have little to contribute to online image GPS location estimation. So, we remove these images from the crawled dataset by checking their average luminance and texture energy. If the image has a high enough or low enough average energy, or has very low texture energy, then it is viewed as noise and removed from the dataset. The texture energy is measured here by the HWVP feature here. For the 170-D feature, we built a matrix of $N * M$, where N is the number of images and M is the number of texture feature dimensions. For each dimension, we computed the maximum value in the matrix denoted as Maxi $(1 * M)$. For the 170D features of each image, each value was divided by its corresponding dimension's maximum value. By doing so, the feature was normalized. Then, the texture energy was calculated by adding all values in the image's features together. We observed that texture of images of low quality was either too high or too low. Instead of giving a hard numeric value for filtering, we deleted alpha% images with too high/low texture energy, respectively, to filter the noisy images. We found this was effective for filtering noisy images when alpha chosen as 1.

B. Feature Representation

In contrast to IM2GPS [2], in this paper, global and local features are utilized to improve the estimation performance and reduce computational costs. Here, color moment (CM) [3] and hierarchical wavelet packet descriptor (HWVP) [4], [38] are used as the global features and SIFT as the local features [5].

1) 45-D Color Moment (CM): Color features have been proven to be the most GPS-informed features [2]. Thus, we use the color feature as the global feature representation of the images in our approach. An image is divided into four equal blocks and a centralized image of equal size. For each block, a 9-D color moment is computed, and thus the dimension of the

color comment for each image is 45. The 9-D color moment of an image segment is utilized, which contains the mean, standard deviation, and skewness of each channel in the HSV color space. The mean E_k, standard deviation σ_k and skewness ω_k of the k-th channel $(k = 1, 2, 3)$ are expressed as follows:

$$E_k = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} P(i, j, k) \tag{1}$$

$$\sigma_k = \left(\frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(P(i, j, k) - E_k \right)^2 \right)^{1/2} \tag{2}$$

$$\omega_k = \left[\frac{1}{H \times W} \sum_{i=1}^{W} \sum_{j=1}^{H} \left(P(i, j, k) - E_k \right)^3 \right]^{1/3} \tag{3}$$

where $P(i, j, k)$ is the value of the pixel located at (i, j) in the k-th channel of a color image.

2) 170-D Hierarchical Wavelet Packet Descriptor (HWVP): Texture feature, also a global description of an image, has been proven to work well for scene categorization and image recognition [3]. We use a hierarchical wavelet packet descriptor (HWVP) [4], [38], a kind of texture feature representation approach, as another global feature in our approach. A 170-D HWVP descriptor is utilized by setting the decomposition level to three and the wavelet packet basis to DB2.

3) Scale Invariant Feature Transform (SIFT): The images could be further described via the local interest point descriptors given by SIFT [5]. The SIFT-based local feature matching is used for assuring: 1) whether or not the input image was taken from an offline GPS location, and 2) which place the input image was taken. In this paper, SIFT feature matching is utilized in both the offline system for representative images selection for each GPS location refined centroid and the online system to determine the matched representative images.

C. Global Feature Clustering

In this paper, we propose clustering the image dataset and using the centroids in the online system instead of the whole image set. Through image clustering, the whole dataset can be divided into sequential small-scale groups according to the appearance of the images. Our main purpose in clustering is to reduce the computational cost and improve the GPS location estimation performance.

K-means clustering has been proven to be a good method for dividing a dataset into small clusters[19]. In this paper, we use it to cluster the global features. To support fast, online GPS estimation, the number of first layer clusters R in k-means should not be set too large. In this paper, R is set according to the different appearances (in color and texture) of images in four seasons, daytime and night, landmark and landscape, modern and ancient. Thus we set $R = 32$. The impact of R to image GPS estimation performances and computational costs is discussed in Section VI-D.

The global feature clustering is carried out on the combined 215-D low-level feature including 45-D color moment and 170-D hierarchical wavelet packet. The global features of all the images in the offline dataset are grouped into R centroids using K-means. After the global feature clustering, we get R centroids $C_1, \ldots, C_R$. Each centroid $C_i (i = 1, \ldots, R)$ is featured by a 215-D global feature vector LC_i.

D. GPS Location Based Cluster Refinement

After obtaining the set of centroids $C_1, \ldots, C_R$, we then partition the set of geo-tagged images into these R clusters. Assuming that the number of the GPS locations in C_i is $N_i (i = 1, \ldots, R)$, we then separate the cluster referring to the GPS locations within each of these clusters, yielding a further partitioning of the images into clusters $c_{i,j} (i = 1, \ldots, R, j = 1, \ldots, N_i)$. For each of these finer clusters, we compute a global feature vector $Lc_{i,j}$ by averaging the global features of the images belonging to the cluster:

$$Lc_{i,j} = \frac{1}{n_{i,j}} \sum_{k=1}^{ni,j} L_{i,j,k}, \; i = 1 \ldots, R; j = 1, \ldots, N_i \tag{4}$$

where $n_{i,j}$ is the image number in $c_{i,j}$, and $L_{i,j,k}$ is the 215-D global feature vector of the k-th image of $c_{i,j}$. Thus the image number z_i in the cluster C_i is the total number of images in each GPS location refined centroids, i.e.,

$$z_i = \sum_{j=1}^{N_i} n_{i,j}; \quad i = 1, \ldots, R; j = 1, \ldots, N_i. \tag{5}$$

Therefore, the total image number Z of the geo-tagged image set is the sum of image numbers in the first layer clusters, i.e.,

$$Z = \sum_{i=1}^{R} z_i; \; i = 1, \ldots, R; j = 1, \ldots, N_i. \tag{6}$$

E. Representative Images Selection for the GPS Location Refined Centroids

The advantage of hierarchical global feature clustering is a low computational cost. In the hierarchical global feature clustering stage, we group images into coarse clusters C_i and refine them into GPS locations refined centroid $c_{i,j}$.

Ideally, the images in the same GPS refined centroid have similar visual content, but actually there are some outliers with incorrect GPS information. Moreover some of the centroids may contain too many images, especially for famous places such as the Eiffel Tower and the leaning tower of Pisa. Thus, selecting representative images for each GPS location, the refined centroid $c_{i,j}$ is helpful for reducing computational costs of online GPS estimation and protecting our approach from the influence of images with faulty GPS information. The impact of using representative images or all the images on image GPS estimation performance and computational costs is discussed in Section VI-D.

In representative image selection, both the relevance of the image to the GPS location and the diversity among representative images are taken into account. The relevance of an image to the GPS location refined centroid is determined by counting the number of matched images from the same cluster. Thus,

we utilize a local feature matching based approach to determine the representative images for each GPS location refined centroid $c_{i,j}$. Only when two images have sufficient matched SIFT point pairs are they considered a match [7], [36]. The diversity is achieved by selecting images representing various viewpoints. The representative images $\{\Omega_1, \ldots, \Omega_l\}$ for each GPS location refined centroid $c_{i,j}$ are determined iteratively, as shown in Algorithm 1.

Algorithm 1 Selecting Representative Images For the GPS Location Refined Cluster $c_{i,j}$

Input:

All the images in $c_{i,j}$ denoted set $\boldsymbol{D}$

Initial:

Pair-wise **match between every pair of** images in $\boldsymbol{D}$;

Determine the number of images matching each image in $\boldsymbol{D}$ by counting the number of matching SIFT features.

Remove images without matched image from $\boldsymbol{D}$;

$A \leftarrow$ the image with most matched images in $\boldsymbol{D}$;

Update: $l \leftarrow 1, \Omega_l \leftarrow A, \boldsymbol{D} \leftarrow \boldsymbol{D} - A$

while $\boldsymbol{D}$ is not null

$A \leftarrow$ the image with most matched images in $\boldsymbol{D}$;

$P_A \leftarrow$ Number of SIFT features in A;

for $k = 1 : l$

Count the number of matched SIFT point n_k between image A and image Ω_k;

end

$P_* \leftarrow \max\{n_1, \ldots, n_l\}, \ * \leftarrow \arg\max\{n_k\}$

if $P_* > P_A/2$

then image A can be viewed as near duplicate with image Ω_*

update: $\boldsymbol{D} \leftarrow \boldsymbol{D} - A$

otherwise image A is assigned as a representative image for the centroid,

update: $l \leftarrow l + 1; \Omega_l \leftarrow A; \boldsymbol{D} \leftarrow \boldsymbol{D} - A$

end

Output: representative images $\{\Omega_1, \ldots, \Omega_l\}$ for the GPS location refined centroid $c_{i,j}$

F. Bow for Representative Images

In this paper, an inverted file structure is proposed for management of the offline dataset, as shown in Fig. 1. First, we randomly sample the SIFT feature points from an image set of about 30 million images, and group the SIFT points into Q centroids (i.e., the BoW number is Q) using a hierarchical K-means based approach. Then for the offline dataset, each SIFT point is quantized into one of the Q centroids.

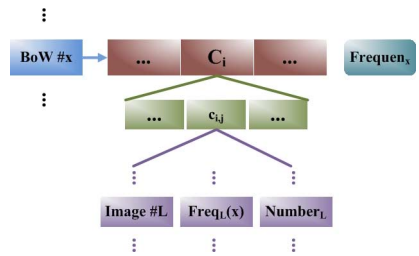


Fig. 2. Inverted file structure for the offline image set. Ci is the i-th first cluster and ci,j denotes the j-th GPS location refined cluster from Ci. In the IFS, Frequenx is the frequency of BoW #x in whole image set. Image #L is the L-th image in the image dataset. The whole number of BoW in image #L is NumberL and the frequency of BoW #x in image #L is $\mathrm{Freq}L(x)$.

After selecting representative images for each GPS location refined centroid, two approaches are proposed to express the GPS location refined centroid $c_{i,j}$. One approach is the normalized histogram (NH) of the BoW of the representative images [35] as follows:

$$NH_{i,j}(k) = \frac{1}{l}\sum_{m=1}^{l} H_{i,j}^{m}(k), k = 1, \ldots, Q \tag{7}$$

where $H_{i,j}^{m}(k)$ is the BoW histogram of the m-th representative image of the refined centroid $c_{i,j}$, and l is the number of the representative images $\{\Omega_1, \ldots, \Omega_l\}$ of the refined centroid $c_{i,j}$.

The other approach is to build an inverted file structure for all the representative images in the offline dataset. The inverted file is also a hierarchical structure as shown in Fig. 2. As for the BoW $\#x(x \in \{1, \ldots, Q\})$, the first layer cluster C_i, the second cluster $c_{i,j}$ and the image $\#L$ it belongs to are all recorded. In addition, the frequency of the BoW in all the image datasets (denoted as $\mathrm{Frequen}_x$) and that in image $\#L$ (denoted as $\mathrm{Freq}_L(x)$) are also recorded. Considering that different images contain a various number of BoW, the number of BoW in image $\#L(\mathrm{Number}_L)$ is recorded as well. All this information is used in the online system. The impact of Q on image GPS estimation performance and computational costs is discussed in Section VI-D.

V. The Online System

The online system estimates the GPS location of an input image with the help of the offline system. The detailed GPS estimation for an input image is shown in Fig. 1. First, we extract the global features and SIFT features for the input image, and quantize the SIFT descriptors into BoW. Then, we carry out GPS estimation using hierarchical clusters selection, local feature matching, and candidate GPS ranking. The hierarchical clusters selection consists of two steps: first layer cluster selection and second layer cluster selection. Local features are used to refine the results. If there is no image matched with the input image (i.e., the best matched images also have very local matching scores), then it is viewed as not taken at any places in the training dataset.

A. First Layer Cluster Selection

Let L_{input} denote the 215-D global features of the input image. The distance D_i between the query image and feature vector LC_i of the i-th center C_i is computed as follow:

$$D_i = \|LC_i - L_{input}\|, (i = 1, \ldots, R) \tag{8}$$

where $\|X\|$ denotes the norm of X. By ranking the distances in ascending order, we select several first layer cluster candidates.

In first layer candidate clusters selection, we aim to choose clusters that have similar texture patterns and colors with the input image. In this paper, the top ranked $M(M \leq R)$ centroids are selected. The reason why we choose M clusters rather than the most similar one is based on the fact that images with the same GPS location may be scattered into different clusters in the first layer, and the visual similarity cannot guarantee the content is the same. Thus, by selecting M clusters in the first layer, it is more likely to find the accurate clusters in the second layer. Let $\mathbb{S} = \{S_i, \ldots, S_M\}$ denote the selected M candidates, where $\mathbb{S}_k \in \{C_i, \ldots, C_R\}$ is one of the selected candidates ($k \in \{1, \ldots, M\}$). The impact of M on image GPS estimation performance and computational cost is discussed in Section VI-D.

B. Second Layer Clusters Selection

After selecting the first layer cluster candidates $\mathbb{S} = \{S_1, \ldots, S_M\}$, the input image can be further refined into the second layer GPS location refined centroids. Each $S_k \in \{C_i, \ldots, C_R\}$ has N_k refined global centroids in the second layer. Thus there are a total of $N = \sum_{k=1}^{M} N_k$ refined centroids for the second layer after selecting M coarse centroids in the first layer. Let $\mathbb{s} = \{r_1, \ldots, r_n\}$ (with $r_i \in \{c_{j,k}\}, j = 1, \ldots, N_j$) denote the set of candidate centroids in the second layer, from which more precise clusters can be determined by the distances d_i between the input image (with its global feature L_{input}) and that of the second layer centroids r_i (with global feature vector Lr_i) as follows:

$$d_i = \|Lr_i - L_{\text{input}}\|, i \in \{1, \ldots, N\}. \tag{9}$$

In the second layer refined clusters selection, we firstly rank the distances d_i in ascending order, and then select the top $V\%$ of the centroids as candidate GPS for the input image. Thus, the number of selected centroids in second layer is $F = V \times N/100$. We denote the selected candidates as $SC = \{g_1, \ldots, g_F\}$ with $g_f \in \{r_1, \ldots, r_N\} (f \in \{1, \ldots, F\})$. The impact of V on image GPS estimation performances and computational costs is discussed in Section VI-D.

C. Local Feature Refinement

The above-mentioned cluster candidate selection is made mainly for the sake of speeding up the process, which does not ensure estimation accuracy. Thus, local feature matching is utilized to improve GPS location estimation performance. As representative images for each second cluster have already been selected in the offline system, we carry out local feature matching for the input image with the representative images of the selected candidates $SC = \{g_1, \ldots, g_F\}$.

In this paper, two different ways are utilized in local feature matching to measure the similarity of the input image and the representative images. One is based on the BoW histograms [35] and the other is based on the inverted file structure of BoW extracted from the representative images in each refined centroid.

1) Bow Histogram Based Similarity Measurement: In the offline system, the normalized BoW histograms of the representative images in each refined centroid are built. Assuming that the BoW histogram of the input image is denoted as $h(k), k = 1, \ldots, Q$, then the similarity of the input image with the refined centroids $c_{i,j}$ can be measured by using cosine similarity (denoted as COS), mean absolute distance (denoted as MAD), mean squared distance (denoted as MSD), and histogram intersection (denoted as HIST) as follows:

$$COS(i,j) = \frac{\sum_{k=1}^{Q} NH_{ij}(k) \times h(k)}{\sqrt{\left[\sum_{k=1}^{Q} (NH_{ij}(k))^2\right] * \left[\sum_{k=1}^{Q} (h(k))^2\right]}} \tag{10}$$

$$MAD(i,j) = \sum_{k=1}^{Q} |NH_{i,j}(k) - h(k)| \tag{11}$$

$$MSD(i,j) = \sum_{k=1}^{Q} (NH_{i,j}(k) - h(k))^2 \tag{12}$$

$$HIST(i,j) = \sum_{k=1}^{Q} \min(NH_{i,j}(k), h(k)). \tag{13}$$

2) Inverted File Structure Based Similarity Measurement: For each BoW that occurs in the input image, we use the obtained inverted files to compute the matching scores of the BoW to the images in the selected candidates $SC = \{g_1, \ldots, g_F\}$. The score is computed while considering the frequency and the weight of BoW by utilizing the well-known Term Frequency-Inverse Document Frequency (TF-IDF) technique. The score of the representative image #L to the input image (denoted as $\text{Score}(L)$) is assigned as the sum of the scores of all the BoW. The score of each image is computed as follows:

$$Score(L) = \sum_{x=1}^{Q} \frac{\omega_x * \text{Freq}_L(X)}{\text{Number}_L * \text{Frequen}_x} \tag{14}$$

where $\text{Freq}_L(x)$ is the frequency of BoW #x and Number_L is the number of BoW in image #L. Frequen_x is the frequency of BoW #x in the whole dataset. ω_x is the weight of BoW #x in the input image.

$$\omega_x = \frac{\text{Freq}_{\text{input}}(x)}{\text{Number}_{\text{input}}} \tag{15}$$

where $\text{Freq}_{\text{input}}(x)$ denotes the frequency of BoW #x and $\text{Number}_{\text{input}}$ is the number of BoW in the input image.

D. GPS Estimation and Verification

The score of each image is used rank the result and to estimate the GPS location. In this paper, a K-NN based approach is utilized in GPS estimation for the input image. From Fig. 1, it

is very likely that images taken from a certain place can be distributed into different clusters due to the various appearances of the images taken at different time and viewpoints. For example, in the second layer, images of the Eiffel Tower are divided into different clusters. So in the online system, K-NN is necessary for improving the GPS location estimation performance. The impact of K on GPS location estimation performance and computational cost are discussed Section VI-D. This approach is similar to that utilized in IM2GPS [2]. First, we use mean-shift clustering for the GPS locations of the images of the selected top ranked K representative images. Finally, we pick out the cluster with the highest cardinality and assign its GPS coordinates to the input image.

To make sure whether or not the estimated GPS of a given input image is taken from the offline locations, we further match the input image with the representative images of the GPS location refined second layer centroid. If two images have sufficient matched SIFT point pairs[7], [36], they are considered a match. Otherwise they are not match. However, the feature matching is comparatively computationally intensive.

Actually in the inverted file structure based approach, the matching scores of the representative images to the input images are obtained as shown in (14). Thus, we can determine whether the GPS of an input image is in the range of the offline dataset or not from the matching scores. In this paper, we first determine the maximum matching scores of the input image to the representative images in the selected candidates $SC = \{g_1, \ldots, g_F\}$, denoted as MScore, and then compute the average score for the selected candidates $SC = \{g_1, \ldots, g_F\}$, denoted as AScore. If the input image has images taken at the same place of interest in the offline dataset, the maximum score should be relatively higher than the average score. On the other hand, if the input image does not have images taken at the same place in the training set, the scores will be low for all the representative images in the selected candidates $SC = \{g_1, \ldots, g_F\}$. Thus the rate of MScore to AScore can be utilized as the standard to judge whether or not the GPS of the input image can be estimated. In this paper, if the rate of MScore to AScore is lower than 1.2, we conclude the GPS of the input image is out of the offline system.

VI. Experiments and Discussions

In order to test the performance of the proposed GPS location estimation approach, we compare IM2GPS [2], a spatial coding based approach (denoted as SC) [20], an SVM based landmark classification method (denoted as LC)[26], and ours. We tested on three datasets: COREL5000 [44], OxBuild5000 [22], and GOLD [35]. Moreover, in order to show its effectiveness for a large-scale dataset with more GPS locations, we also tested our approach on GOLDEN, which we built. **GOLDEN** is a large-scale geo-tagged image set with 5.2 M images from 1,447 places of interest all over the world. GOLDEN was also crawled from Flickr. Cross validations between GOLD and GOLDEN are also presented to show the robustness of the proposed approach. All the experiments were performed in a C environment on a server with 2.0 GHz CPU and 24 GB memory.

A. Experimental Dataset

The categories of **OxBuild5000** and **COREL5000** serve as GPS locations. Thus, the GPS numbers for OxBuild5000 and COREL5000 are 14 and 50 respectively. 100 images were selected randomly from the whole dataset as the test set, while the remaining images served as the training set in the offline system for the construction of the hierarchical structure.

GOLD contains more than 3.3 million images with their Geotags. It was crawled from Flickr using its public API. 80 travel spots were selected for testing, i.e., the number of GPS locations was 80. The test dataset for the 80 sites contained 52,046 images [35]. A more detailed description of GOLD is provided in APPENDIX.

GOLDEN is a large-scale, geo-tagged image set with 5.2 M images from 1,447 places of interest all over the world. GOLDEN was also crawled from Flickr. The selection of 1,447 places is referred to the list of places of interest all over the world from WIKI.com. There is no content overlapping between GOLD and GOLDEN. That is to say, the GPS locations in GOLD do not appear in GOLDEN, and vice is versa.

B. Performance Evaluation

The performance evaluation contains two parts. The first part is to test the cross validation performances that utilize images taken outside the GPS locations in offline systems as input. The second part is to test the average recognition rate of test images taken from the GPS locations in offline systems.

1) Error Recognition Rate: We use error recognition rate (ER) to evaluate the cross validation performances. For an image not taken in any of the places in the offline dataset, if it is determined to be one of the GPS locations in the offline systems (i.e., the matching scores of the input image with the representative images are large enough), then we judge the GPS estimation is wrong. The ER is expressed as follows:

$$ER = \frac{FN}{TN} \times 100\% \tag{16}$$

where TN is the number of test images selected for cross validation and FN is the number of images wrongly estimated as GPS locations taken from the offline system.

2) Average Recognition Rate: As for the test images taken from places in the offline systems, if the selected image group is actually the same group from which the test image is from, it is correctly estimated. Otherwise, it is falsely estimated. We use average recognition rate (AR) to evaluate the GPS estimation performance, which is given as follows:

$$AR = \frac{1}{G} \sum_{i=1}^{G} A_i \tag{17}$$

where A_i is the correct recognition rate of the i-th spot

$$A_i = \frac{NC_i}{NA_i} \times 100\%,\ i \in \{1, \ldots, G\} \tag{18}$$

where NC_i is the correct estimated image number, and NA_i is the test image number. G is the total number of GPS locations.

TABLE I
AVERAGE RECOGNITION RATES (%) OF SC(1-NN) AND SC(K-NN), IM2GPS, LC AND OUR APPROACH COS, MAD, MSD, HIST AND IFS ON COREL5000, OXBUILD5000 AND GOLD

Dataset	SC(1-NN)	SC(K-NN)	IM2GPS	LC	COS	MAD	MSD	HIST	IFS
COREL5000	58.64	76.01	45.98	49.43	97.00	96.00	97.00	95.00	91.00
OxBuild5000	40.98	60.87	39.67	53.94	91.00	90.00	90.00	89.00	87.00
GOLD	36.76	71.84	53.06	54.25	84.64	84.05	85.02	84.21	83.94

TABLE II
AVERAGE COMPUTATIONAL COSTS (IN MS) OF SC(1-NN) AND SC(K-NN), IM2GPS, AND OUR APPROACH COS, MAD, MSD, HIST AND IFS ON COREL5000, OXBUILD5000 AND GOLD

Dataset	SC(1NN)	SC(K-NN)	IM2GPS	LC	COS	MAD	MSD	HIST	IFS
COREL5000	7.30	7.94	60.46	1.04	0.76	0.71	0.82	1.08	0.07
OxBuild5000	5.51	5.42	33.74	1.34	0.47	0.41	0.50	0.49	0.09
GOLD	39.60	47.00	64927	2.89	0.96	0.93	1.03	0.99	0.16

C. GPS Estimation Performance Comparisons

For a fair comparison, only visual features of the input image are utilized. As for SC [20], both 1-NN and K-NN are applied in the comparisons, which are denoted by SC(1-NN) and SC(K-NN), respectively. In SC(K-NN), K is set to be 120 under which best performance is achieved in our experiments. As for IM2GPS, we use the best parameters provided in the paper [2]. As for LC[26], the size of BoW is set at 60K. The performance of our approach under the similarity measurement methods COS, MAD, MSD, HIST, and IFS are evaluated. The parameters in our baseline algorithm are set at $R = 32, M = 10, K = 50, V = 100$, and the size of BoW is set at 60K. The GPS estimation performance of SC(1-NN), SC(K-NN), IM2GPS, LC, COS, MAD, MSD, HIST and IFS are shown in Table I. The corresponding computational costs are shown in Table II.

The results show that our method can achieve a significantly better performance than the other methods, not only in GOLD but also in both OxBuild5000 and COREL5000. The average precisions of IM2GPS on the three-test dataset are 45.98%, 39.67%, and 53.06%. The average precisions of LC are 49.43%, 53.94%, and 54.25%, respectively. The performances of SC (K-NN) in the three test dataset are 76.01%, 60.87%, and 71.84%, which achieves performance improvements over their corresponding SC(1-NN). The results for our method under COS for the three datasets are 97%, 91%, and 84.64%, respectively. Our approach under MAD, MSD, HIST, and IFS performs better than IM2GPS, LC, and SC.

We have found that both global and local features are beneficial in image GPS estimation. Because IM2GPS utilizes only global features, its AR is comparatively low. Although SC utilizes local features, it neglects the clues that global features can provide. Thus, our method can achieve a better performance. There are two reasons for the relatively low recognition rates for LC. One is that spatial information is somewhat neglected in their using of the BoW histogram. The other is that SVM classifiers are affected by the outliers (images with incorrect GPS information) in training.

The average computational costs of IM2GPS on the three test sets are 60.46 micro-second (ms), 33.74ms, and 64927ms, while that of SC (K-NN) are 7.30ms, 5.51ms, and 39.60ms on the three test sets respectively. The LC is also time efficient with its computational costs 1.04ms, 1.34ms, and 2.89ms, respectively. The computational costs of COS, MAD, MSD, HIST, and IFS are all lower than SC, LC, and IM2GPS. For the large-scale dataset, our approach under IFS is very efficient and the average computational cost is 0.117ms, which is only about 1.8×10^{-6} of IM2GPS, 0.25% of SC(K-NN), 12.19% of COS, 12.58% of MAD, 11.36% of MSD, and 11.82% of HIST.

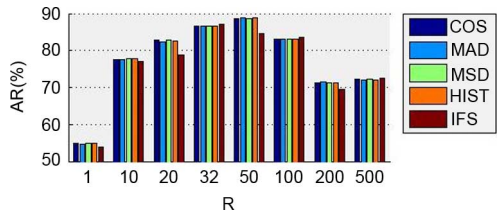


Fig. 3. Impact of first layer cluster number R to GPS estimation performance.

D. Discussions

The performance of our approach is related to five parameters: the number of first layer clusters R, the number of first layer candidates M, the percentage of the selected second layer cluster candidates V, K in K-NN, and the size of the SIFTS descriptor codebook. The parameters in our baseline algorithm are set at $R = 32, M = 10, K = 50, V = 100$, and the size of BoW is set at 60K. We will next examine their respective impacts by carrying out a set of experiments on GOLD. Finally, the impact of using representative images and all the images for each refined centroid to the GPS estimation performances is also provided.

1) Impact of Total Number of the First Layer Clusters R: To study the impact of the total number of the first layer, we carry out experiments under different R by fixing $V = 100, M = R$, and $K = 50$. The corresponding Average Recognition rates of COS, MAD, MSD, HIST, and IFS with $R = \{1, 10, 20, 32, 50, 100, 200, 500\}$ are shown in Fig. 3, and their computational costs are shown in Table III respectively. In Fig. 3, $R = 1$ means that no global clustering is utilized. As R increases, the performance first increases and then drops. When R is in the range of $[20, 100]$, a better performance

TABLE III
AVERAGE COMPUTATIONAL COSTS (MS) UNDER DIFFERENT R

R	1	10	20	32	50	100	200	500
COS	0.15	0.92	4.17	10.08	25.24	90.17	358.97	3209.18
MAD	0.13	0.86	3.47	8.62	21.64	86.07	342.81	3143.65
MSD	0.18	0.99	4.5	11.04	25.76	91.09	387.97	3412.86
HIST	0.16	0.94	4.29	10.46	25.39	90.49	367.73	3341.08
IFS	0.06	0.07	0.10	0.19	0.18	0.20	0.18	0.19

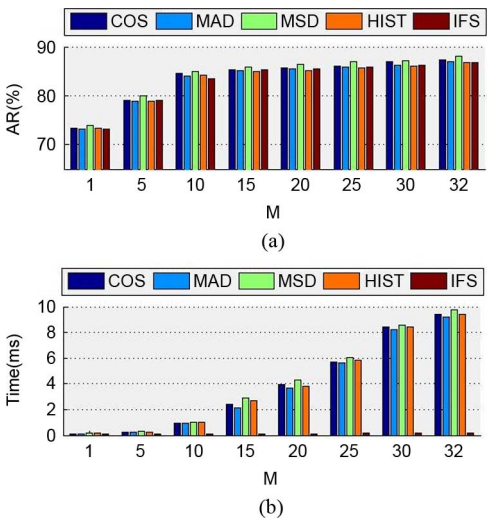


Fig. 4. Impact of first layer candidate M to GPS estimation performances. (a) AR values of COS, MAD, MSD, HIST and IFS, (b) the computational costs (ms) under various M.

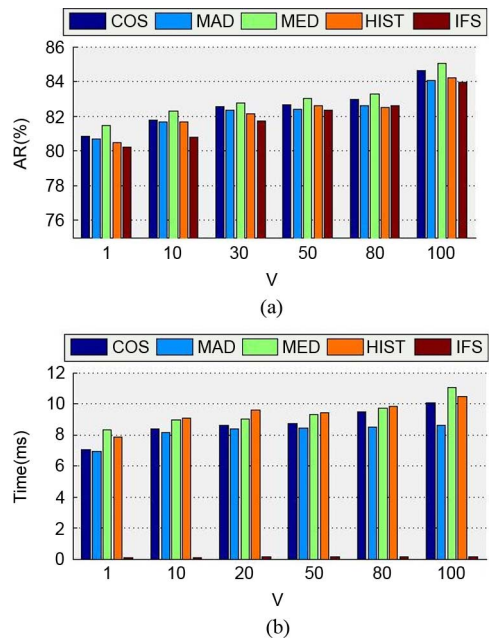


Fig. 5. The impact of the percentage of second layer candidate V to GPS estimation performances. (a) AR values of COS, MAD, MSD, HIST and IFS, (b) computational costs.

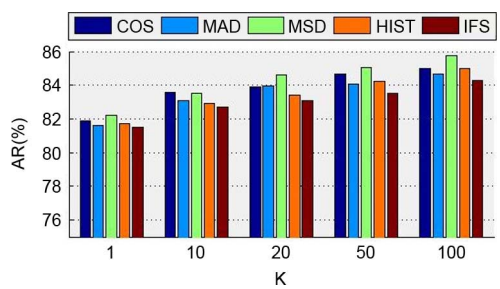


Fig. 6. Impact of K in K-NN to GPS estimation performance.

can be achieved. As shown in Table III, as R increases, the computational costs of COS, MAD, MSD, HIST increase dramatically, while the computational costs of IFS are all less than 0.20 ms.

2) Impact of Number of First Layer Candidate M: Fig. 4(a) shows the AR values with the increase of $M(M \leq R)$, when $R = 32, K = 50$, and $V = 100$. Fig. 4(b) shows the computational costs with the increase of M. It is clear that with the increase of M, the computational costs increase lineally for COS, MAD, MSD, and HIST. However, the computational costs of IFS do not change very much with the increase of M. IFS is very efficient compared to COS, MAD, MSD, and HIST.

3) Impact of Percentage of Second Layer Candidate V: The impact of the percentage of second layer candidate V on GPS estimation performance is tested under the condition that $R = 32$, $M = 10$, and $K = 50$. The AR values and the corresponding computational costs (ms) are shown in Figs. 5(a) and 5(b) respectively. When $V = 1, V = 10, V = 30, V = 50, V = 80$, and $V = 100$, the corresponding AR values are 80.85%, 81.78%, 82.55%, 82.63%, 82.97%, and 84.81%, respectively, for COS, and 80.07%, 80.56%, 81.43%, 82.35%, 82.59%, and 83.97% for IFS. The performances of COS, MAD, MSD, and HIST are a little better than IFS. The computational costs increase with the increase of V for COS, MAD, MSD, and HIST. The computational costs of IFS are much more stable and far less than those of COS, MAD, MSD, and HIST.

4) Impact of K in K-NN: Next, we discuss the impact of K, which is the parameter of the last step of GPS estimation when $R = 32, M = 10$, and $V = 100$. The corresponding AR values of COS, MAD, MSD, HIST, and IFS when $K = \{1, 10, 20, 50, 100\}$ are shown in Fig. 6. When $K = 1, K = 10, K = 20, K = 50$ and $K = 100$, the corresponding AR values are 81.91%, 83.68%, 83.91%, 84.64%, and 84.97% for COS, and 81.53%, 82.88%, 83.31%, 83.94%, and 84.27% for IFS. The results show that the AR for all the methods improves with the increase of K. As K is used in the last step for image GPS estimation, the computational costs under various K are almost the same.

5) Impact of BoW Size Q: In the above sections, experiments were conducted on the condition that the BoW size was set at

TABLE IV
AR (%) OF OUR APPROACHES UNDER DIFFERENT SIZES OF BOW (Q)

Dataset	Size of BoW Q	COS	MAD	MED	HIST	IFS
COREL5000	6K	87.83	86.97	88.01	87.54	81.23
	30K	92.11	92.06	92.22	91.10	86.68
	60K	97.12	96.05	**97.40**	95.30	91.01
	300K	**97.24**	**96.42**	96.99	**95.41**	**95.54**
OxBuild5000	6K	88.97	87.07	89.05	86.13	85.95
	30K	91.33	89.42	88.83	87.43	86.41
	60K	91.01	90.02	91.04	89.01	87.23
	300K	**91.35**	**90.31**	**91.05**	**90.10**	**89.52**
GOLD	6K	83.04	81.10	82.43	82.03	81.48
	30K	84.06	83.29	84.92	83.31	82.55
	60K	84.64	84.05	85.02	84.21	83.94
	300K	**85.04**	**84.87**	**85.43**	**84.62**	**86.52**

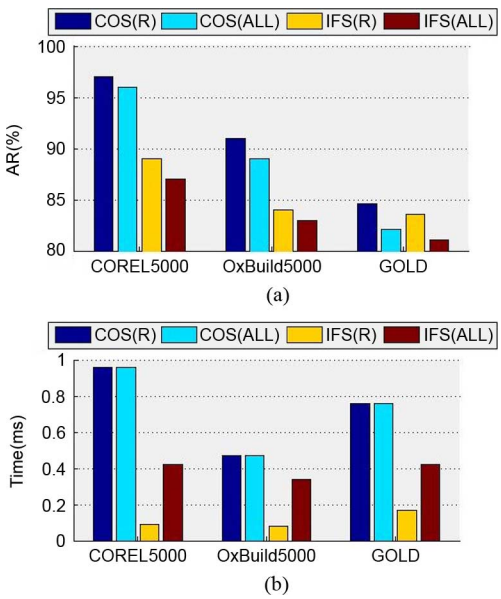


Fig. 7. Impact of using representative images vs. all images on COREL5000, OxBuild5000 and GOLD. (a) AR values of COS(R), COS(ALL), IFS(R), and IFS(ALL). (b) Computational costs.

60K. Here, the impact of BoW size on the GPS location estimation is discussed by setting it to 6K, 30K, 60K, and 300K when $R = 32$, $M = 10$, $V = 100$, and $K = 50$. The results are shown in Table IV. It can be observed that usually, the larger the BoW size, the better the performance of GPS location estimation.

6) Impact of Using Representative Images or All Images: In the offline system, representative images are selected for every GPS location refined centroid. The aim of selecting representative images is to reduce the influence of noise geo-tagged images on GPS estimation performance and computational cost. Here, comparison between using representative images and using all images in each refined centroids is discussed. Let COS(R) and COS(ALL) denote our approach COS by using representative images and all images for input image GPS location estimation. And let IFS(R) and IFS(ALL) denote IFS utilizing representative images and all images, respectively.

TABLE V
AR (%) OF IFS ON GOLDEN WITH DIFFERENT R AND M

M	R=10	R=32	R=50
2	64.20	79.85	76.41
32	--	81.03	79.43
50	--	--	80.24

TABLE VI
AVERAGE COMPUTATIONAL COSTS (MS) OF IFS ON GOLDEN WITH DIFFERENT R AND M

M	R=10	R=32	R=50
2	0.27	0.48	1.75
32	--	1.24	3.69
50	--	--	10.82

The corresponding AR values and their computational costs are shown in Figs. 7(a) and 7(b), respectively. From Fig. 7(a), we can see that the performances when using representative images are better than those using all images on all three test datasets COREL5000, OxBuild5000, and GOLD. The computational cost of COS is not affected by using a representative image at all, because the average BoW histograms of all images and representative images have the same dimension. IFS can save a lot time using representative images instead of using all images. The computational cost of IFS using representative images is about a quarter of that using all images.

E. Subjective GPS Estimation Result

In order to show the subjective GPS estimation result, 21 images randomly selected from the 80 spots are assigned to the estimated GPS location as shown in Fig. 8. We mark the estimated GPS locations of the photos with solid lines. We use the green lines to show the correct estimation and the red lines to show incorrect estimation (their correct GPS locations are shown by the dashed green lines). It can be observed that our method works well for landmarks. On the other hand, for some landscapes, the performance is not satisfactory because the representative images have changing backgrounds depending upon the time. For example, the test images ***Cape of Good Hope*** and ***Mount Fuji***, which are marked by red frames, are not correctly estimated. The photo of ***Mount Fuji*** is crowded with flowers and grasses.

Fig. 8. Subjective results of GPS estimation performances for some test samples. Red line indicates wrong estimation while green solid line denotes right estimation and green dash line denotes the accurate location. The images with red frames are the wrong estimations.

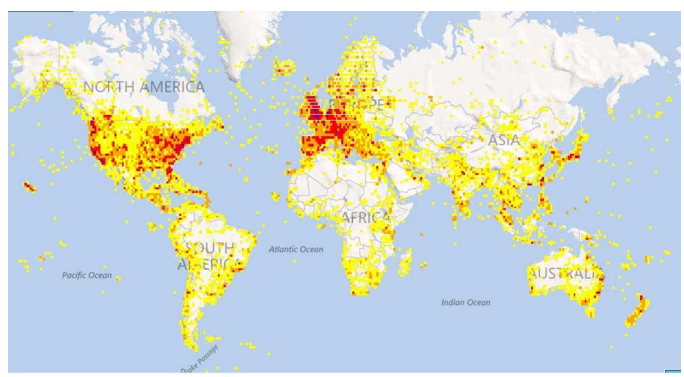
There are too many SIFT points in the background, and only a few on the mountain, so local feature refinement cannot improve the GPS estimation performance.

F. Performance Evaluation on GOLDEN

To test the scalability of our method IFS, an experiment on the **GOLDEN** dataset is provided. The experiment was implemented on GOLDEN with a different R and $M(M \leq R)$ when $K = 50$ and $V = 100$. The AR values and computational costs of IFS are given in Tables V and VI, respectively. It is clear that our method also achieved a good performance on the extended test dataset. The AR was 81.03% and the computational cost was 1.24 ms when $R = 32$ and $M = 32$. When $R = 32$ and $M = 2$, the AR is 79.85% and the computational cost is only 0.48 ms. When $R = 50$, and $M = 2$, the performance decrease was about 3.5%, and the computational cost increased by about three times. Thus, it is reasonable to set $R = 32$ in global feature clustering. The experiment showed that even for a dataset with more GPS locations, IFS has a satisfactory performance and acceptable computational cost.

G. Cross Validation Between GOLD and GOLDEN

In order to show the effectiveness of the proposed approach in image GPS estimation, cross validations between GOLD and GOLDEN were carried out to accomplish the task of recognizing images taken outside of the GPS location in the offline system. The performances when using GOLD as the training set and images in GOLDEN as the test set are presented in Table VII. We randomly selected 1000, 2000, 3000, 5000, and 10000 images from GOLDEN as input images, the results of incorrectly judged image numbers are 72, 145, 204, 336, and 671, respectively. Correspondingly, the performance when using



(a)

(b)

Fig. 9. Crawled image distributions and the selected 80 travel spots. (a) the image distribution in the world scale, (b) the distribution of the 80 travel spots. In Fig. 9(a), purple indicates the places contain more than 5000 images, and light gray means the number of the images ranges from 3000 to 5000, red from 2000 to 3000, green from 1000 to 2000, orange 500 to 1000 and yellow is from 100 to 500.

GOLDEN as the training set and images in GOLD as the test set are presented in Table VIII. In the cross validation, when

TABLE VII
ERROR RATES (%) OF IFS USING GOLD AS TRAINING SET AND RANDOMLY SELECTED IMAGES FROM GOLDEN AS TEST SET

Number of Test Image	1000	2000	3000	5000	10000
FN	72	145	204	336	671
ER	7.2%	7.25%	6.8%	6.72%	6.71%

TABLE VIII
ERROR RATES (%) OF IFS USING GOLDEN AS TRAINING SET AND RANDOMLY SELECTED IMAGES FROM GOLD AS TEST SET

Number of Test Image	1000	2000	3000	5000	10000
FN	84	172	241	382	751
ER	8.4%	8.60%	8.03%	7.64%	7.51%

TABLE IX
INDEX OF 80 TRAVEL SITES

Spot #	Spot Name	Spot #	Spot Name	Spot #	Spot Name
0	Forbidden City	27	Colosseum	53	Fa Men Temple
1	Taj Mahal	28	Venice, Italy	54	Mausoleum of the First Qin
2	Angkor Wat	29	Parthenon, Greece	55	Mount Hua
3	Bali	30	Red Square in Moscow	56	Stele Forest
4	Borobudur,	31	Big Ben	57	Tang Paradise
5	Sentosa	32	Buckingham Palace	58	Xi'an mosque
6	Crocodile Farm	33	London Tower Bridge,	59	Suez Canal
7	Pattaya Beach	34	Westminster Abbey	60	Tiananmen Square
8	Babylon	35	Monte Carlo	61	Great Hall of the People
9	Mount Fuji	36	Niagara Falls	62	Monument to the People's Heroes
10	Aswan High Dam	37	Honolulu	63	Summer Palace
11	Nairobi National Park	38	Yellowstone	64	Ruins of the Old Summer Palace
12	Cape of Good Hope	39	Statue of Liberty	65	Peking Man Site
13	Pyramids	40	Times Square	66	Ming Dynasty Tombs
14	The Nile, Egypt	41	Central Park	67	Lugou Bridge
15	Great Barrier Reef	42	Yosemite National Park	68	Prince Gong Mansion
16	Sydney Opera House	43	Grand Canyon	69	Beijing Ancient Observatory
17	Ayers Rock	44	Hollywood	70	Temple of Heaven
18	Mount Cook	45	Disneyland	71	Temple of Earth
19	Easter Island	46	Miami	72	Temple of Sun
20	Notre Dame de	47	Metropolitan Museum of Art	73	Temple of Moon
21	Eiffel Tower	48	Acapulco	74	Lama Temple
22	Arch of Triumph	49	Cuzco, Mexico	75	Bei Hai Park
23	Elysee Palace	50	Bell tower in Xi'an	76	Jing Shan Park
24	Louvre, France	51	Great Wild Goose Pagoda	77	Confucius Temple
25	Kolner Dom	52	drum tower in Xi'an	78	Fragrant Hills
26	Leaning Tower of Pisa			79	Grandview Garden

we use randomly selected 1000, 2000, 3000, 5000, and 10000 images from GOLD as input images, the resulting error rates are 8.4%, 8.6%, 8.03%, 7.64%, and 7.51%, respectively. This shows that our method can accomplish the task of recognizing images taken out of offline locations.

VII. CONCLUSIONS

In this paper, we propose a system of hierarchical structure to estimate the GPS location for an image. Both GPS estimation performances and computational costs are beneficial from the hierarchal structure and inverted file structure. The hierarchical global feature clustering divides the large-scale geotagged dataset into a set of small-scale clusters. The heavy computing costs of local feature matching is reduced dramatically by confining local feature match to several small-scale GPS location refined clusters. Utilizing representative images rather than all the images of each GPS location helps further save computational costs and improve the GPS estimation accuracy. The inverted file structure has also proved to be efficient in GPS estimation, especially for a large-scale image dataset. Our approach works well for estimating GPS locations for landmarks. However, it is very challenging to estimate image GPS for photos taken from places of interest with a changing background. In the future, we will pay more attention to this.

APPENDIX

The crawled data set GOLD are from Flickr, the images with tags such as 'birthday', 'party', 'meeting' and so on which have little to do with location are deleted automatically. After the preprocessing, we get an image dataset with GPS information containing 3.3 million (3M) images.

Then GPS distribution of the images in GOLD is as shown in Fig. 9(a). The 3.3M images are distributed in 652912 different

places. And different image number is annotated in different color. In Fig. 9(a), purple indicates the places contain more than 5000 images, and light gray means the number of the images ranges from 3000 to 5000, red from 2000 to 3000, green from 100 to 2000, orange 500 to 1000 and yellow is from 100 to 500. Finally, 80 travel spots are selected by considering both the image number and the user number of who upload the images of each location. Table IX shows the name of the 80 travel spots. Fig. 9(b) shows the locations of the 80 spot in map.

REFERENCES

[1] M. A. Stricker and M. Orengo, "Similarity of color images," in *Proc. IS&T/SPIE's Symp. Electron. Imaging: Sci. & Technol.*, 1995, pp. 381–392.

[2] J. Hays and A. A. Efros, "IM2GPS: Estimating geographic information from a single image," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2008, pp. 1–8.

[3] B. S. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 837–842, 1996.

[4] X. Qian, G. Liu, D. Guo, Z. Li, Z. Wang, and H. Wang, "Object categorization using hierarchical wavelet packet texture descriptors," in *Proc. 11th IEEE Int. Symp. Multimedia*, 2009, pp. 44–51.

[5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, pp. 91–110, 2004.

[6] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, pp. 1615–1630, 2005.

[7] L. S. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," in *Proc. 17th Int. Conf. World Wide Web*, 2008, pp. 297–306.

[8] W. B. Thompson, C. M. Valiquette, B. H. Bennet, and K. T. Sutherland, "Geometric reasoning for map-based localization," in *Comput. Sci. Tech. Rep. UUCS-96–006*. Salt Lake City, UT, USA: Univ. Utah, 1996.

[9] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3D," in *Proc. ACM Trans. Graph. (TOG)*, 2006, pp. 835–846.

[10] I. Simon, N. Snavely, and S. M. Seitz, "Scene summarization for online image collections," in *Proc. IEEE 11th Int. Conf. Comput. Vision*, 2007, pp. 1–8.

[11] T. Quack, B. Leibe, and L. Van Gool, "World-scale mining of objects and events from community photo collections," in *Proc. Int. Conf. Content-Based Image and Video Retrieval*, 2008, pp. 47–56.

[12] A. Popescu and P. Moëllic, "MonuAnno: automatic annotation of georeferenced landmarks images," in *Proc. ACM Int. Conf. Image and Video Retrieval*, 2009, p. 11.

[13] J. Kleban, E. Moxley, J. Xu, and B. S. Manjunath, "Global annotation on georeferenced photographs," in *Proc ACM Int. Conf. Image and Video Retrieval*, 2009, p. 12.

[14] K. Yang, M. Wang, X. Hua, and H. Zhang, "Social image search with diverse relevance ranking," in *Proc. Advances in Multimedia Modeling*, 2010, pp. 174–184.

[15] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Trans. Multimedia Comput., Commun., Applicat. (TOMCCAP)*, vol. 2, pp. 1–19, 2006.

[16] T. Quack, U. Mönich, L. Thiele, and B. S. Manjunath, "Cortina: A system for large-scale, content-based web image retrieval," in *Proc. 12th Annu. ACM Int. Conf. Multimedia*, 2004, pp. 508–511.

[17] Y. Rui, T. S. Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in mars," in *Proc. Int. Conf. Image Processing*, 1997, pp. 815–818.

[18] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *J. Roy. Statist. Soc. Series C (Appl. Statist.)*, vol. 28, pp. 100–108, 1979.

[19] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian, "Spatial coding for large scale partial-duplicate web image search," in *Proc. Int. Conf. Multimedia*, 2010, pp. 511–520.

[20] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian, "Building contextual visual vocabulary for large-scale image applications," in *Proc. Int. Conf. Multimedia*, 2010, pp. 501–510.

[21] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vision*, 2003, pp. 1470–1477.

[22] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2007, pp. 1–8.

[23] C. Wu, F. Fraundorfer, J. Frahm, and M. Pollefeys, "3D model search and pose estimation from single images using VIP features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition Workshops*, 2008, pp. 1–8.

[24] G. Baatz, K. Köser, D. Chen, R. Grzeszczuk, and M. Pollefeys, "Leveraging 3d city models for rotation invariant place-of-interest recognition," *Int. J. Comput. vision*, vol. 96, pp. 315–334, 2012.

[25] M. Park, J. Luo, R. T. Collins, and Y. Liu, "Beyond GPS: Determining the camera viewing direction of a geotagged image," in *Proc. Int. Conf. Multimedia*, 2010, pp. 631–634.

[26] Y. Li, D. J. Crandall, and D. P. Huttenlocher, "Landmark classification in large-scale image collections," in *Proc. IEEE 12th Int. Conf. Comput. Vision*, 2009, pp. 1957–1964.

[27] D. J. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg, "Mapping the world's photos," in *Proc. 18th Int. Conf. World Wide Web*, 2009, pp. 761–770.

[28] E. Kalogerakis, O. Vesselova, J. Hays, A. A. Efros, and A. Hertzmann, "Image sequence geolocation with human travel priors," in *Proc. IEEE 12th Int. Conf. Comput. Vision*, 2009, pp. 253–260.

[29] Y. Zheng *et al.*, "Tour the world: Building a web-scale landmark recognition engine," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, 2009, pp. 1085–1092.

[30] R. Ji, Y. Gao, B. Zhong, H. Yao, and Q. Tian, "Mining Flickr landmarks by modeling reconstruction sparsity," *ACM Trans. Multimedia Comput. Commun., Applicat. (TOMCCAP)*, vol. 7, pp. 31–31, 2011.

[31] R. Ji, X. Xie, H. Yao, and W. Ma, "Mining city landmarks from blogs by graph modeling," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 105–114.

[32] R. Adam and P. Kelm, "Working notes for the placing task at mediaeval," 2012.

[33] M. Trevisiol, J. Delhumeau, H. Jégou, and G. Gravier, "How INRIA/IRISA identifies geographic location of a video," in *Proc. MediaEval 2012 Workshop Working Notes*, 2012.

[34] O. Van Laere, S. Schockaert, and B. Dhoedt, "Ghent University at the 2011 placing task," in *Proc. MediaEval Workshop Working Notes*, 2011.

[35] J. Li, X. Qian, Y. Y. Tang, L. Yang, and C. Liu, "GPS estimation from users' photos," *Adv. Multimedia Model.*, pp. 118–129, 2013.

[36] Y. Xue and X. Qian, "Visual summarization of landmarks via viewpoint modeling," in *Proc. 19th IEEE Int. Conf. Image Processing (ICIP)*, 2012, pp. 2873–2876.

[37] X. Qian, X. Liu, C. Zheng, Y. Du, and X. Hou, "Tagging photos using users' vocabularies," *Neurocomputing*, 2013.

[38] X. Qian, D. Guo, X. Hou, Z. Li, H. Wang, and G. Liu, "HWVP: Hierarchical wavelet packet descriptors and their applications in scene categorization and semantic concept retrieval," *Multimedia Tools Applicat.*, pp. 1–24, 2012.

[39] O. Van Laere, S. Schockaert, and B. Dhoedt, "Ghent University at the 2010 placing task," in *Proc. MediaEval 2010 Workshop*, 2010.

[40] L. Tzy Li, J. Almeida, D. Carlos Guimaraães Petronette, O. A. B. Penatti, and R. da. S. Torres, "A multimodal approach for video geocoding at mediaeval 2012," in *Proc. MediaEval 2012 Workshop*, 2012.

[41] P. Kelm, S. Schmiedeke, and T. Sikora, "Video2GPS: Geotagging using collaborative systems, textual and visual features," in *Proc. MediaEval 2010 Workshop*, 2010.

[42] X. Li, C. Hauff, M. Larson, and A. Hanjalic, "Preliminary exploration of the use of geographical information for content-based geo-tagging of social video," in *Proc. MediaEval 2012 Workshop*, 2012.

[43] P. Kelm, S. Schmiedeke, and T. Sikora, "How spatial segmentation improves the multimodal geo-tagging," in *Proc. MediaEval 2012 Workshop*, 2012.

[44] G. Liu, Z. Li, L. Zhang, and Y. Xu, "Image retrieval based on microstructure descriptor," *Pattern Recognit.*, vol. 44, p. 2123, 2011.

[45] H. Liu, T. Mei, J. Luo, Li Houqiang, and Li Shipeng, "Finding perfect rendezvous on the go: Accurate mobile visual localization and its applications to routing," in *Proc. ACM Multimedia*, 2012, pp. 9–18.

Jing Li received the B.A. degree from Xi'an Jiaotong University in 2010, and now is a Master student in SMILES lab, Xi'an Jiaotong University.

Her research interests include computer vision, large scale image retrieval and recognition and data mining and knowledge discovery from social multimedia.

Xueming Qian (M'10) received the B.S. and M.S. degrees in Xi'an University of Technology, Xi'an, China, in 1999 and 2004, respectively, and the Ph.D. degree in the School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an, China, in 2008. He was awarded Microsoft fellowship in 2006. From 1999 to 2001, he was an Assistant Engineer at Shannxi Daily. From 2008 until now, he is a faculty member of the School of Electronics and Information Engineering, Xi'an Jiaotong University. Now he is an associate professor of the School of Electronics and Information Engineering, Xi'an Jiaotong University. He is the director of SMILES LAB. He was a visiting scholar at Microsoft Research Asia from Aug. 2010 to March 2011. His research interests include video/image analysis, indexing, and retrieval.

Yuan Yan Tang (F'04) is a Chair Professor in Faculty of Science and Technology at University of Macau and Professor/Adjunct Professor/Honorary Professor at several institutes including Chongqing University in China, Concordia University in Canada, and Hong Kong Baptist University in Hong Kong. His current interests include wavelets, pattern recognition, image processing, artificial intelligence. He has published more than 400 academic papers and is the author/coauthor of over 25 monographs/books/book chapters. He is the Founder and Editor-in-Chief of International Journal on Wavelets, Multiresolution, and Information Processing (IJWMIP), and Associate Editors of several international journals. He is the Founder and Chair of pattern recognition committee in IEEE SMC. He has serviced as general chair, program chair, and committee member for many international conferences. Dr. Tang is the Founder and General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition (ICWAPRs). He is the Founder and Chair of the Macau Branch of International Associate of Pattern Recognition (IAPR). Dr. Y. Y. Tang is a Fellow of IEEE, and Fellow of IAPR.

Linjun Yang is a Researcher with Microsoft Research Asia, Beijing, China. He received the MS degree in Fudan University, Shanghai, China, in 2006. His current research interests include multimedia information retrieval and computer vision.

Tao Mei (M'07–SM'11) is a Lead Researcher with Microsoft Research Asia, Beijing, China. He received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively. His current research interests include multimedia information retrieval and computer vision. He has authored or co-authored over 150 papers in journals and conferences, eight book chapters, and edited three books. He holds eight U.S. granted patents and more than 20 in pending.

Dr. Mei was the recipient of several paper awards from prestigious multimedia conferences, including the Best Paper Awards at ACM Multimedia in 2007 and 2009, the Best Poster Paper Award at the IEEE MMSP in 2008, the Top 10% Paper Award at the IEEE MMSP in 2012, the Best Paper Award at ACM ICIMCS in 2012, the Best Student Paper Award at the IEEE VCIP in 2012, and the IEEE TRANSACTIONS ON MULTIMEDIA Prize Paper Award 2013. He received Microsoft Gold Star Award in 2010, and Microsoft Technology Transfer Awards in 2010 and 2012. He is an Associate Editor of Neurocomputing and the Journal of Multimedia, a Guest Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE Multimedia Magazine, the ACM/Springer Multimedia Systems, and the Journal of Visual Communication and Image Representation. He is the Program Co-Chair of MMM 2013, and the General Co-Chair of ACM ICIMCS 2013. He is a Senior Member of the IEEE and the ACM.