Topic Models for Image Localization

Zheng Wang
Faculty of Science
University of Ontario Institute of Technology
Oshawa ON L1H 7K4 Canada
zheng.wang@uoit.ca

Faisal Z. Qureshi
Faculty of Science
University of Ontario Institute of Technology
Oshawa ON L1H 7K4 Canada
faisal.qureshi@uoit.ca

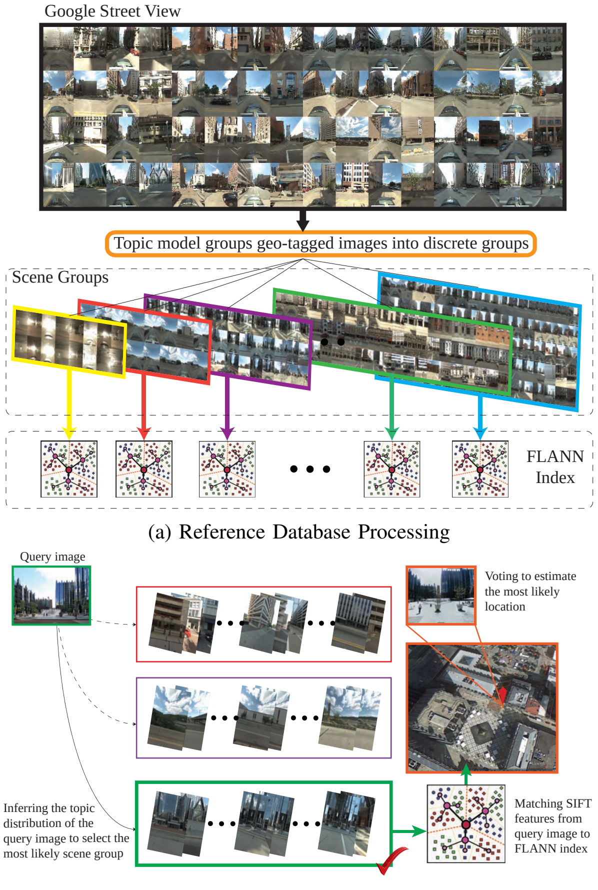
***Abstract*—We present a new scheme for partitioning geo-tagged reference image database in an effort to speed up (query) image localization while maintaining acceptable localization accuracy. Our method learns a topic model over the reference database, which in turn is used to divide the reference database into scene groups. Each scene group consists of "visually similar" images as determined by the topic model. Next raw Scale-Invariant Feature Transform (SIFT) features are collected from every image in a scene group a Fast Library for Approximate Nearest Neighbours (FLANN) index is constructed. Given a query image, first its scene group is determined using the topic model and then its SIFT features are matched against the corresponding FLANN index. The query image is localized using the location information from the visually similar images in the reference database. We evaluate our approach on Google Map Street View dataset and demonstrate that our method outperforms a competing technique.**

***Keywords*-image localization; topic models; SIFT; FLANN; visual words;**

I. Introduction

The ability to geo-localize an image is an important enabling capability [1]. Of the billions of images stored in the cloud–Flickr, Smugmug, Facebook, etc.—many are already geo-tagged. These geo-tagged images constitute a vital source of data for such applications that require the exact location (e.g. Global Positioning System coordinates) of an image for subsequent processing. Agarwal *et al.*, for example, construct a 3D model of Rome from a set of geo-tagged images [1]. A vast sum of images available in the cloud are not geo-tagged, however. Still it is possible to geo-localization images using existing geo-tagged images by exploiting visual similarities between the query images and one or more geo-tagged images. Zamir and Shah presented a system capable of using low-level image features to geo-localize an image using Google Maps Street View data [2].

Finding visually similar geo-tagged images to geo-localize a query image is akin to Content Based Image Retrieval (CBIR). The fundamental step is to find the set of visually similar geo-tagged images from the reference set and use spatial filtering to assign the most


Google Street View
Topic model groups geo-tagged images into discrete groups
Scene Groups
FLANN Index
Query image
Voting to estimate the most likely location
Inferring the topic distribution of the query image to select the most likely scene group
Matching SIFT features from query image to FLANN index

(a) Reference Database Processing

(b) Query Image Localization

Figure 1: The proposed approach. (a) A sampling of images taken from Google Maps Street View Dataset for Pittsburg, PA, which serve as our geo-tagged reference database. The reference database is partitioned into scene groups and a SIFT FLANN index is constructed for each scene group. Partitioning into scene groups is carried out by learning a topic model over the reference database. The topic model groups visually similar images into one group. (b) Given a query image, topic model is used to identify the most relevant scene group. Next raw SIFT features from the query image are matched against the FLANN index for this scene group to localize the query image. (*FLANN index visual representation courtesy of D.A. Miro.)*

likely location to the query image. Given that we are dealing with geo-tagged data comprising hundreds of thousands of images per city, any proposed scheme must scale gracefully. Currently the dominant approach is to organize the low-level features (typically, SIFT or some variation thereof) computed from the reference geo-tagged data into an index that supports fast nearest neighbour queries. Such an index, for example, can be constructed using Fast Library for Approximate Nearest Neighbours (FLANN) [3]. Features computed from the query image are then matched against those stored in the index to identify a set of images that are "visually similar" to the query image. Next spatial filtering is used to estimate the location for the query image [2].

In our experience constructing a single index that contains features from every geo-tagged image in a reference database (say Google Maps Street View) does not scale. An obvious solution to address this scalability limitation is to divide the reference database into different sets and construct an index for each set. Features from the query image can then be processed simultaneously at each index, improving query times and addressing scalability issues. In this paper, however, we have take a different approach. We begin by constructing a topic model for our reference database. Topic model is used to divide the reference dataset into *scene groups*. Images within a scene group have similar visual characteristics as determined by the topic model. Next for each scene group we build a FLANN index over SIFT features extracted from the images belonging to that scene group. Given a new image that needs to be localized, we first infer its scene group and then use the corresponding FLANN index to determine the set of visually similar geo-tagged images. This set is then used to assign a location to the query image [2]. An overview of the proposed system is shown in Fig. 1. The proposed approach is able to achieve significant speed increase over the technique described in [2] while achieving similar localization accuracy.

Our approach is motivated by our observation about how an individual might geo-localize an image. When presented with an image a human may look for landmarks that might help him narrow the search space. For instance, if presented with an image showing CN Tower[1], the individual will immediately focus his attention to other images of Toronto. Scene groups, we find, mimic this behavior.

Topic models initially appeared for statistical analysis of textual documents [4]. Recently these have been applied to analyze image data. Specifically topic models have been used develop image annotation [5], [6], scene classification and modeling [7], [8], [9] and even retrieval [10] schemes. To the best of our knowledge, ours is the first application of topic models to image localization.

Our reference dataset comprises more than $50,000$ images from Google Map Street View Dataset for Pittsburg, PA. We compare our approach against the method proposed by Zamir and Shah in [2]. Our method outperforms their method in terms query processing times, while achieving similar localization accuracies.

A. *Overview*

The remainder of the paper is organized as follows. We briefly outline the related work in the next section. Sec. III describes our scheme of processing the reference database and partitioning it into scene groups. In the following section we present the query processing. Sec. V presents the results and we conclude the paper with Sec. VI.

II. Related Work

Early methods in image localization focused on estimating camera position and orientation under the assumption that both the query image and the set of reference images are dominated by scenes of easily recognizable buildings [11], [12]. Here image similarity is determined by matching outlines (or templates) of buildings appearing in the query and the reference images. [11], for example, constructs reference templates by identifying the facade of a building. Each template corresponds to the dominant plane of the building and consists of conspicuous edges. Robertson and Cipolla [12] use building outlines and vanishing points to extract "canonical view" from original views in order to eliminate the 3D orientation of the camera with respect to the building [12]. These methods, while promising, lack a mechanism to efficiently match a query image to the set of reference images.

Recently image geo-localization techniques are increasingly relying upon the existence of a reference set of geo-tagged images. Zamir and Shah [2] gather SIFT features from the geo-tagged reference set and build a FLANN index over these features. In their scheme matched SIFT features are pruned using both 1) real-world distances between the query image and the reference images that contain the matched SIFT features and 2) the distance between the SIFT features in feature space.

General purpose CBIR techniques such as using Bag-of-Words model and Vocabulary-Trees to match the query image against the set of geo-tagged images is also used to collect visually similar geo-tagged images [13]. Bag-of-Words and Vocabulary trees, while efficient, alone perhaps are not the right choice for image localization as these suffer from *polysemy*[2] and

[1] CN Tower is a Toronto, ON landmark.

[2] A single visual word may be present in images taken at different physical locations.

synonymy [9].[3] Hays and Efros construct a hybrid feature comprising such features as tiny images, color histograms, etc. to match the query image against the set of geo-tagged reference images [14]. Their approach lacks an efficient index for matching the query image to the reference set. As a consequence their approach does not scale.

Torii *et al.* [15] observe that the query image usually consists of visual words from several nearby reference images. They find the location of the query image by describing it as a linear combination of locations of nearby images containing the same visual words as the query image. Kalantidis [16] suggests dividing the reference images into different sets based upon their locations. These sets are called *scene map*. A vocabulary tree is then constructed for each set. It is not immediately obvious how this scheme speeds up localization or improve localization accuracy.

Our method also aims to partition the reference set; however, we do not rely upon the locations of the images. Rather we learn a topic model from the reference images and use topic distributions for different images to partition the images into different sets.[4] Our conjecture is that partitioning the reference set based upon locations only is not the right approach. Consider for example two images take at the same location but in different directions. These two images most likely will have very different visual content (Fig. 2). Furthermore we have also noticed that partitioning images using topic model tend to group visually similar images—say images of some landmark in a city—together. Which in turn may lead more memory efficient indexing schemes.

The proposed method aims to address the shortcomings of both feature based approaches and bag-of-words and vocabulary tree schemes. Feature based approaches are accurate; however, these suffer from scalability issues. On the other hand bag-of-word and vocabulary tree schemes can efficiently match the query image with the reference image set; however, these schemes exhibit high localization errors. In a sense ours method combines these two approaches: topics models are used to partition the reference dataset into scene groups and a feature-driven approach is used to match the query image within the most relevant group. The proposed method simultaneously address accuracy and scalability concerns.

III. Processing Geo-Tagged Dataset

The proposed scheme builds an index over the reference database consisting of geo-tagged images. This index is used to efficiently match a query image against the reference database in order to collect visually similar geo-tagged images. We now describe the various steps of this process.

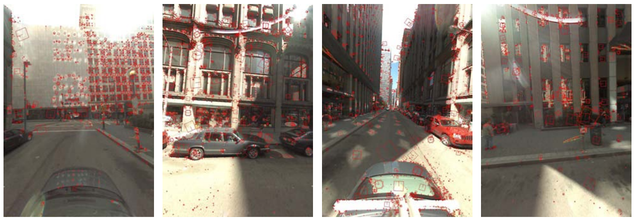


Figure 2: SIFT features detected on 4 images from the Google Maps Street View data. These four images are taken at the same location in four cardinal directions.

A. Collecting Visual Words

The first step is to collect SIFT features from all the images in the reference database (Fig. 2). Each SIFT feature $\mathbf{f}$ is geo-tagged. Say $\mathbf{f}$ represents a 128 dimensional SIFT vector that appeared in an image $\mathbf{I}$ taken at a location l (*latitude* and *longitude*). Then we represent a geo-tagged SIFT feature as a tuple $\langle \mathbf{f}, l \rangle$. Our reference dataset yields around 28.3 million SIFT features. We are currently using Google Map Street View dataset for the city of Pittsburg, PA, which consists of $50,220$ images. Clearly it is infeasible to construct an efficient index over such a large number of SIFT features.

We use K-means to perform vector quantization over raw SIFT features. After this process we are left with $2,792,343$ visual words. Each word $\mathbf{w}$ is a SIFT feature denoting the center of one of the clusters returned by the vector quantization process. Each reference image $\mathbf{I}_j$ is described by a collection of visual words $\mathcal{I}_j = \{\mathbf{w}_0, \mathbf{w}_1, \mathbf{w}_2 \cdots \mathbf{w}_k\}$. In the next section we will train the topic model on the collection of visual words aggregated over all reference images.

B. Learning a Topic Model

Topic model sees each image as a random mixture of (latent) topics, which in turn is responsible for generating the visual words associated with that image. The goal of learning a topic model is to find the latent topic set that can best explain the visual words observed in each image. We employ Latent Dirichlet Allocation (LDA) generative model to learn the topic model over the aggregated set of visual words observed in our reference dataset [18].

When learning a topic model we have a choice to make about the dimensionality of our topic space. A higher dimensional topic space can explain fine features within an image; whereas, a lower dimensional topic space cannot. At the same time, however, a higher dimensional topic space can suffer from over-fitting, meaning that the topic model is easily distracted by visual clutter present in an image. Such a topic model

[3] Different visual words may describe the same scene.

[4] Topic models is a vast area of research and we refer the gentle reader elsewhere for more information about this subject [7], [17].

will not generalize. We have experimented with different dimensions for the topic space. In this paper we present results for topic dimensions of 100 and 300 (see Sec. V).

Given a topic model each image $\mathbf{I}_j$ is mapped onto the topic space. Here each image is represented as a T-dimensional vector $[z_0, z_1, z_2, \cdots, z_T]$, where T is the dimensionality of the topic space and z_i is the ith component. An intuitive way to look at the topic space is to consider each of its dimensions as a particular *item (or topic)* seen in the image. E.g., a dimension may refer to the existence of a Green patch in the image. Each image is simply a linear combination of different topics.

C. Scene Groups

The next step is to group images that have similar topic distributions. We achieve this through another round of clustering. Specifically we cluster topic vectors corresponding to each of the reference image into K groups. We refer to these groups as *scene groups*. The premise is that visually similar images will be grouped together since these will exhibit similar topic distributions. Fig. 3 shows sample images belonging to three different scene groups. Notice how images belonging to a scene group share similar visual features. Notice also how scene groups depicted in Fig. 3(b) and Fig. 3(c) are constructed around unique buildings (or landmarks). A possible explanation for this "spatial partitioning" is that the images of a visually distinctive landmark share many visual characteristics, plus that these images are sufficiently different from all other images. Note also that the topic model fails to create a spatial partitioning for images of nondescript buildings (Fig. 3(a)). This is to be expected.

The proposed method then constructs an index for each scene group. For this purpose it collects raw SIFT features from images belonging to a particular scene group and construct a FLANN index over these SIFT features. Specifically the FLANN index for a given scene group is constructed using geo-tagged SIFT features belonging to the images in that scene group. FLANN index currently does not support online modifications; however, it does support fast approximate nearest neighbour queries.

IV. Localizing a Previously Unseen Image

After processing the reference database, we are left with K FLANN indices, one for each of the scene groups. When presented with a new image that needs to be localized, topic model learned over the reference database is used to infer topic distribution for this new image. The process involves three steps: 1) computing raw SIFT features, 2) projecting the computed SIFT features to the visual word space constructed for the reference database and 3) using the visual words present in the query image to infer its topic distribution.

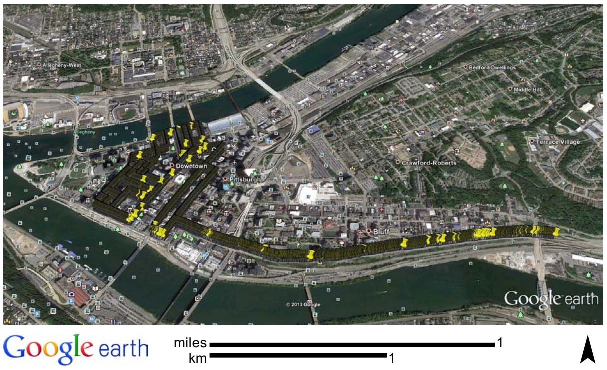


Figure 4: Google Maps Street View dataset for Pittsburg, PA, which serves as our reference dataset, covers the area highlighted above. Currently we are not able to localize images taken outside of this area. Similarly we cannot localize indoor images taken within this area.

The topic distribution is then used to identify the most likely scene group for the query image. Next raw SIFT features from the query image are matched against the FLANN index corresponding to the most likely scene group for that query image. This process identifies the set of matched geo-tagged SIFT features $\{\langle \mathbf{f}_j, l_j \rangle | j > 0\}$.

We adopt the scheme proposed in [2] to prune the set of SIFT features returned after matching the FLANN index of the most likely scene group. Say $\langle \mathbf{f}_i, l_i \rangle$ represents a geo-tagged SIFT feature that appears in the query image and $NN_i(k)$ represents the kth nearest neighbour of this SIFT feature. Also say the location of the kth nearest neighbour is $l_i(k)$. Then we use the following equation to determine whether or not to consider SIFT feature $\mathbf{f}_i$ found in the query image during its localization.

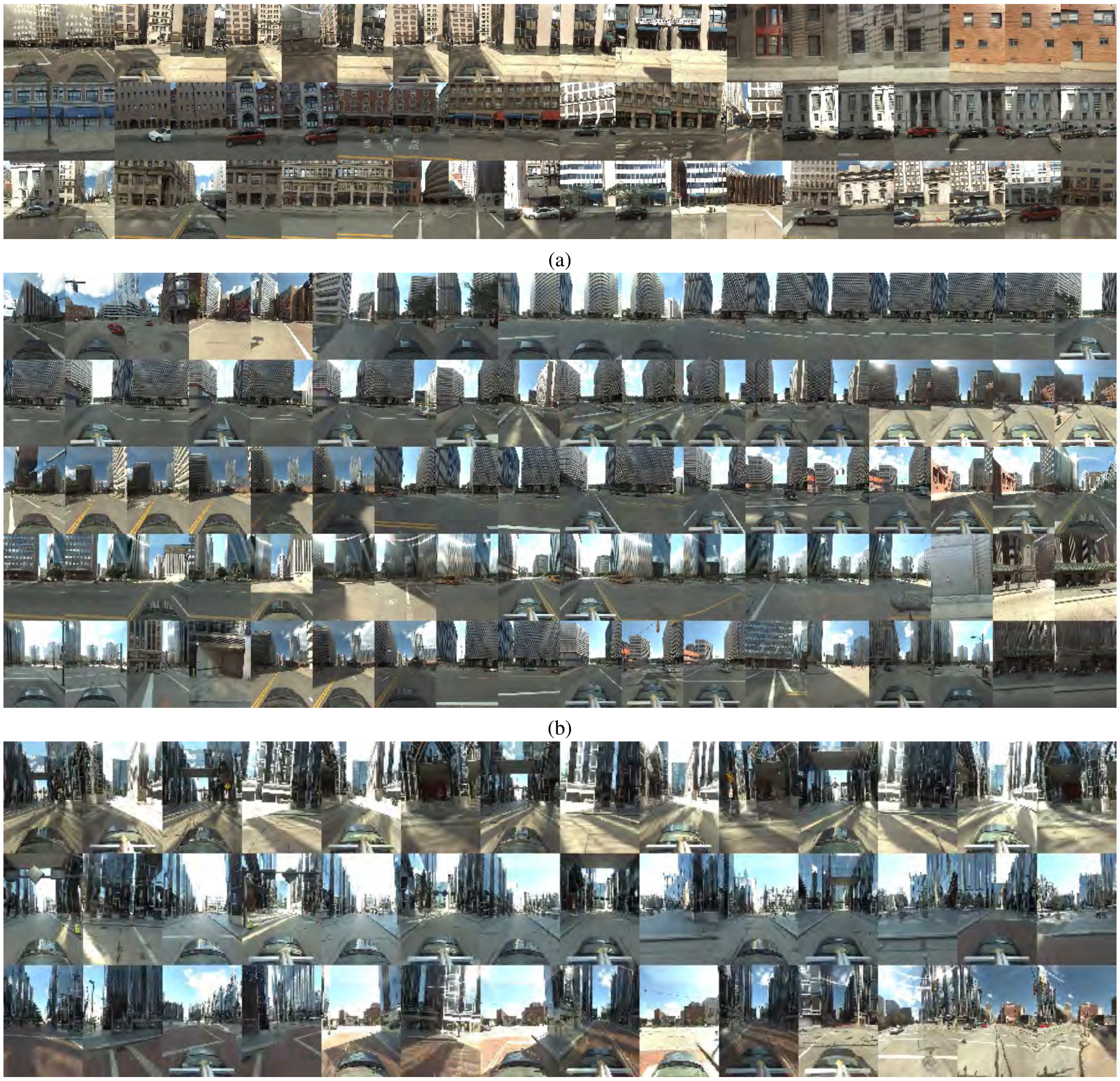
$$u(\mathbf{f}_i) = \begin{cases} 1 & \frac{\|\mathbf{f}_i - NN_i(1)\|}{\|\mathbf{f}_i - NN_i(i, Min(j))\|} < 0.8, \\ & \forall j \leftarrow \|l_i(1) - l_i(j)\| > D \\ 0 & \text{otherwise,} \end{cases}$$

where D is distance threshold.

The matched geo-tagged SIFT features that pass this test are used to assign a location to the query image through voting. Essentially each of the matched feature votes for its location and the location that has the largest number of votes is assigned to the query image.

V. Experiments

We evaluate the proposed method using geo-tagged images from Google Maps Street View data for Pittsburg, PA. It consists of roughly $50,220$ images. Google Maps Street View database consists of an image cube (4 images taken in cardinal directions) taken at 12



(a)

(b)

(c)

Figure 3: A sampling of images belonging to different scene groups. Notice how topic model driven partitioning scheme groups visually similar images together. (a) is a generic scene group. (b) is a scene group comprising images that all see a unique bulding and (c) consists of images that see the famous PPG Place, Pittsburg, PA. (b) and (c) supports the notion that scene groups are constructed around landmarks.

meters interval. We noticed that this dataset exhibits some redundancy. We addressed this issue by only considering image cubes 36 meters apart for our reference database; therefore, our reference database consists of only $16,740$ images. Subsampling the reference database in this manner did not adversely effect the performance of our approach.

Around 28.3 million SIFT features are collected over these $16,740$ images. The raw SIFT features when loaded in RAM takes about 13.3 Gigabyte of space. K-means clustering is used to construct $2,792,343$ visual words from the raw SIFT features. This represents roughly 10 percent of the total number of raw features. Topic model is learned over these visual words. We have experimented with topic space dimensions of 100 and 300. Topic distributions aggregated over all the images are then clustered to construct scene groups and FLANN indices are constructed for each of the scene group.

A higher dimensional topic space is able to capture fine features of an image; however, may suffer from overfitting. Additionally it takes longer to infer the topic distribution of the query image when topic space has a large number of dimensions. Table I, for example, lists the topic distribution inference times (average) for a query image for 100 dimensional and a 300 dimensional topic spaces. The good news is that topic distribution

Dimensions	Sequential (sec)	Parallel (sec)
100	1.1	0.1487
300	3.1139	0.445

Table I: Topic distribution inference times for a query image.

(a) (b)

Figure 5: A sampling of test images used to evaluate our method. These images are already geo-tagged, so we have the ground truth available for these images.

inference times can be drastically reduced by using the parallel algorithms described in [19] (Table I, column 2).

Since the primary thrust of this paper is to partition the reference geo-tagged images into groups of visually similar images by employing the topic model learned over all the images. We have evaluated our approach using topic models consisting of 100 and 300 dimensions. We have also experimented with varying the number of scene groups (10, 50 and 100) that we construct from the reference database.

We have evaluated our approach on 121 test images.[5] The actual GPS locations of these images are know. These serve as our ground truth. GPS locations of the test images were not used during the localization procedure. Fig. 5 shows a subset of our test images. Note that Google Maps Street View data does not contain these images. In other words these images are previously unseen as far as our reference database is concerned.

We divided the test images into two groups. The first group consists of those images of Pittsburg that do not contain any unique (visually distinctive) buildings. We call this set *non-landmark images*. Where as the second group consists of those images of Pittsburg that see some unique buildings, e.g., the PPG Place that dominates Pittsburgh skyline. We refer to this set as *landmark images*. Remembering that we are dividing the reference database into different groups using visual similarity. Our intuition is that reference images that see a particular landmark will be grouped together. To test whether this is indeed the case and to ascertain if there are any benefits to such a grouping, we evaluate the proposed approach on both non-landmark and landmark images. For the tests presented below we assume that an image is correctly localized if it is within 200 meters of its true GPS location. We used a desktop (Intel Core i5 2.8GHz processor, 6GB RAM) running Windows 7 to carry out the experiments presented here.

84 non-landmark images		
Model	Accuracy	Time (s)
Method in [2]	26	14.4136
100-Topic / 10 groups	24	3.352
100-Topic / 50 groups	20	2.3205
100-Topic / 100 groups	19	1.526
300-Topic / 10 groups	**26**	**2.55567**
300-Topic / 50 groups	24	1.6579
300-Topic / 100 groups	16	1.3893

Table II: Localization performance comparison for non-landmark images.

37 landmark images		
Model	Accuracy	Time (s)
Method in [2]	36	11.4019
100 Topics / 10 groups	**36**	**2.33134**
100 Topics / 50 groups	32	1.6241
100 Topics / 100 groups	29	1.01103
300 Topics / 10 groups	33	2.05295
300 Topics / 50 groups	33	1.21132
300 Topics / 100 groups	31	1.1616

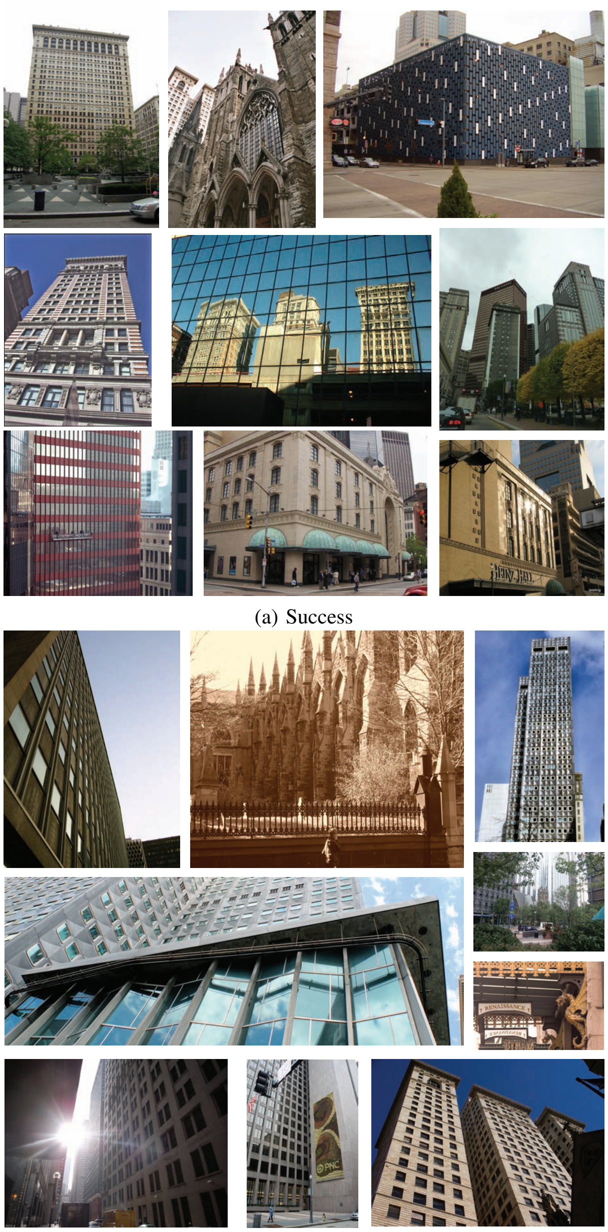
Table III: Localization performance comparison for landmark images.

Table II summarizes the results of our approach using 84 non-landmark images. It also compares our approach to that of [2]. Note that Zamir and Shah's approach is able to correctly localize 26 images out of 84. Our method is also able to correctly localize 26 images (300 topic dimensions; 10 groups). A reason for such low accuracy is that our reference database is impoverished. One needs a lot more than just $50,220$ images to cover a city the size of Pittsburg. Fig. 6(b) depicts a sample of non-landmark images that were localized incorrectly by our method. It is worth mentioning that these images exhibit a high visual dissimilarity with the images present in our reference database

Zamir and Shah's method on average takes roughly 14 seconds to process a query (Table II). Our method is able to match the accuracy of Zamir and Shah's method (300 topic dimensions; 10 groups); however, our method processes a query on average in under 3 seconds. Consequently on average our method processes a query roughly 5 times faster than that of Zamir and Shah's method.

Table III summarizes the results of our approach using 37 landmark images. It also compares our approach to that of [2]. Note that Zamir and Shah's approach is

[5] We got these images from Panoramio [20].



(a) Success

(b) Failures

Figure 6: (a) Some of the non-landmark images that were localized correctly by our method and (b) a selection of non-landmark images that were localized incorrectly by our method.

121 test images		
Model	Accuracy	Time (s)
Method in [2]	62	12.90775
100 Topics / 10 groups	**60**	**2.84167**
100 Topics / 50 groups	52	1.9723
100 Topics / 100 groups	48	1.268515
300 Topics / 10 groups	59	2.30431
300 Topics / 50 groups	57	1.43461
300 Topics / 100 groups	47	1.2755

Table IV: Image localization performance comparison aggregated over landmark and non-landmark images.

able to correctly localize 36 images out of 37. Both our method and the method presented in [2] are able to achieve an accuracy of 97%. Here too our method matches the accuracy of Zamir and Shah's approach; however, on average our method is spends a little over 2 seconds per query and their method takes around 11 seconds to process a query. This is again a 5 fold increase in query processing speed.

It is worth keeping in mind that the localization accuracy for landmark images is much higher than that of non-landmark images. This is to be expected. Images showing generic items are typically much harder to localize without a very detailed reference database. Tables II and III also exhibit a trend that we anticipated. Irrespective of the topic dimensions, partitioning the reference database into more scene groups adversaly affects the accuracy and at the same time increases query processing speed. Increase in the processing times is easy to explain—more scene groups mean fewer SIFT features per group and smaller FLANN indices. Decrease in the accuracy; however, is related to the fact that the chance that the topic model will assign the query image to the wrong group increases as we increase the number of scene groups.

Table IV aggregates the results for both non-landmark and landmark images. Our method compares favorably with the method proposed by Zamir and Shah in terms of accuracy (ours 49.5% compared to theirs 51%). However, on average the proposed method processes a query 4.5 times faster than their method.

VI. Conclusions

We present a method for partitioning a geo-tagged reference image database into scene groups. Images belonging to a scene group share salient visual characteristics as determined by the topic model that is learned over the entire dataset. For examples all images of some landmark may be collected into a single scene group. Next a SIFT-based FLANN index is constructed for each scene group. Given a query image that needs to be localized, first its scene group is inferred using the same topic model that was initially used to partition the reference dataset. Next SIFT features computed from the query image are matched against the FLANN index of the most relevant scene group to identify a set of "visually similar" geo-tagged images. These images are then used to localize the query image.

We have evaluated the proposed method using Google Maps Street View Dataset for Pittsburg, PA. In terms of accuracy our method compares favorably with a competing approach developed by Zamir and Shah [2]. However our method is able to process query images at nearly 4.5 times the rate of the method presented in [2].

We have experimented with topic space dimensions and have found that it does effect the accuracy of the proposed method. In general a low-dimensional topic space better capture landmark images; whereas, a high-dimensional topic is needed when dealing with non-landmark images.

We realize that we have barely scratched the surface and that many more experiments are needed to fully understand the behavior of the system. Still our initial results appear promising. Some ideas for future work might be: 1) how many visual words should be chosen to learn a good topic model for a given reference database? 2) how to dynamically update the topic model as new geo-tagged images arrive, etc.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski, "Building rome in a day," in *In Proc. 2009 IEEE 12th International Conference on Computer Vision*, vol. 2, October 2009, pp. 72–79.

[2] A. R. Zamir and M. Shah, "Accurate image localization based on google maps street view," in *In Proc. of European Conference on Computer Vision*, vol. 6314, Crete, Greece, September 2010, pp. 255–268.

[3] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *International Conference on Computer Vision Theory and Application VISSAPP'09)*. INSTICC Press, 2009, pp. 331–340.

[4] C. H. Papadimitriou, H. Tamaki, P. Raghavan, and S. Vempala, "Latent semantic indexing: a probabilistic analysis," in *In Proc. of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, Seattle, Washington, USA, June 1998, pp. 159–168.

[5] Y. Feng and M. Lapata, "Topic models for image annotation and text illustration," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, ser. HLT '10, June 2010, pp. 831–839.

[6] D. Putthividhya, H. Attias, and S. Nagarajan, "Supervised topic model for automatic image annotation," in *2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, March 2010, pp. 1894–1897.

[7] L. Fei-Fei and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, June 2005, pp. 524–531.

[8] A. Bosch, A. Zisserman, and X. Muñoz, "Scene classification via plsa," in *9th European Conference on Computer Vision*, Graz, Austria, May 2006, pp. 517–530.

[9] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. Van Gool, "Modeling scenes with local descriptors and latent aspects," in *Tenth IEEE International Conference on Computer Vision*, vol. 1, October 2005, pp. 883–890.

[10] E. Hörster, R. Lienhart, W. Effelsberg, and B. Möller, "Topic models for image retrieval on large-scale databases," *ACM Sigmultimedia Records*, vol. 1, pp. 15–16, December 2009.

[11] B. Johansson and R. Cipolla, "A system for automatic pose-estimation from a single image in a city scene," in *In Proc. of International Conference on Signal Processing, Pattern Recognition, and Applications*, Crete, Greece, 2002.

[12] D. Robertson and R. Cipolla, "An image-based system for urban navigation," in *In Proc. of the British Machine Vision Conference*. BMVA Press, 2004, pp. 819–828.

[13] D. Nistr and H. Stewnius, "Scalable recognition with a vocabulary tree," in *In Proc. of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE Computer Society, 2006, pp. 2161–2168.

[14] J. Hays and A. A. Efros, "im2gps: estimating geographic information from a single image," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[15] A. Torii, J. Sivic, and T. Pajdla, "Visual localization by linear combination of image descriptors," in *IEEE International Conference on Computer Vision Workshops*, November 2011, pp. 102–109.

[16] Y. Kalantidis, G. Tolias, Y. S. Avrithis, M. Phinikettos, E. Spyrou, P. Mylonas, and S. D. Kollias, "Viral: Visual image retrieval and localization," *Multimedia Tools and Applications*, vol. 51, no. 2, pp. 555–592, 2011.

[17] T. Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, pp. 50–57.

[18] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty, "Latent dirichlet allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[19] D. Newman, P. Smyth, and M. Steyvers, "Scalable parallel topic models," *Journal of Intelligence Community Research and Development*, August 2007.

[20] Last visited: 13 February 2013. [Online]. Available: http://www.panoramio.com/