Spatial Rerank-based Bag-of-Words Model

Thuyen V. Phan
Advanced Program in Computer Science
Faculty of Information Technology
University of Science, VNU - HCM
Email: pvthuyen@apcs.vn

Minh-Bao Truong
Advanced Program in Computer Science
Faculty of Information Technology
University of Science, VNU - HCM
Email: tmbao@apcs.vn

***Abstract*—The popularity of many social network sites such as Facebook, Twitter, and YouTube creates a huge demand of managing and querying visual data from billions of images. Searching by text cannot describe the information as effective as by images but it is more complicated to search by images due to the changes of camera angles or lighting conditions. To address this problem, the authors conduct a comparison of the retrieval quality between a standard Bag-of-Words model and one with spatial reranking using RANSAC. Our experimental results on Oxford Building 5K Dataset show that spatial reranking improves the mean Average Precision of Bag-of-Words model from 0.676 up to 0.741. The authors view this work as a promising step to further improve the performance of different visual search systems using the spatial relations between the images.**

I. Introduction

In our lives, there are many emotional and memorable moments that worth keeping and sharing with others. Therefore, services allowing users to upload and share personal photos are always one of many notable products of different companies such as Facebook, Flickr, Instagram, and Google Photos. This shows that sharing photos is one of greatest demand of users on the Internet. These services also allow users to attach some memos to their photos as well as to search their photos more easily using text queries.

Currently, the most common way for user to do so is to tag their photos manually which takes users a lot of time and effort. There are also some proposed methods (cite) and smart systems which are able to automatically identify noticeable landmarks or location related to the photos such as Google Photos and Flickr. However, these automated annotation are identical for all users and thus do not reflect one's own memories, feelings or characteristics. For example, these systems would recommend phrases like ”Eiffel Tower”, ”a dog”, or ”a cat” rather than ”where I first met my lover” or the name of your pet. Therefore, it owuld be necessary for a system to automatically tag users' photos with personalized caption with respect to their personal features.

In this paper, we propose to develop a system that can suggest appropriate annotations for each photo uploaded by users using Visual Instance Search. In our system users can assign his/her personalized annotations for some photos as initial examples then, the system will automatically propagate these annotations to other existed photos in their collection based on the visual similarities among photos. For each uploaded photo, the system will base on the visual similarity between the uploaeded photo and already-annotated photos of the corresponding users to identify a list of suitable annotations for the uploaded photo in the descending order of the similarity. Then, users can choose to approve reasonable annotation for the uploaded photo. Inaddition, if a user upload more than one photos and change the annotations, the system will have more samples to reference from and thus, it will be better adapted to the user's interests. Therefore, our system is not only able to recommend proper annotations which are unique for each user but also to interactively learn and adapt as users change the annotations.

Since the problem of retrieving similar images in a collection corresponding to a single image has been developed for years, there are many different approaches that can be used to solve this problem. One of them is template matching method, i.e. a technique for finding small parts of an image which match a template image [3], [4], [5]. Another popular technique is to evaluate the similarity of two images by comparing some regions which seem seem to be the interested point of the images, namely features matching [6], [7], [8]. In our system, the authors develop our own Visual Instance Search framework using Bag-of-Words model. In Bag-of-Words model, each image is represented as a histograms of pre-trained visual words (codebook). Since Bag-of-Words allows parts of a query image to appear flexible in the result images, it is a potential approach that is widely used and and focused by many research groups.

Together with the exponential increasing of the number of uploaded images, the system faces lots of difficulty adapting those new images. Since re-training the codebook requires changing Bag-of-Words vector of users' existing images and takes too much time, the authors propose to use a fixed codebook trained with different types of images (e.g. cars, dogs, cats, buildings...) and use it universally. Because of the varieties of those different images, it is appropriate to compute and represent any new images' Bag-of-Words vector without changing the codebook. We trained our codebook using ABC dataset tested our system on XYZ dataset. Our performed experiments show that ...

Our main contributions in this paper are as follows:

- First, we propose the idea and realize the system that can recommend annotation for photos with visual instance search.

- Second, our system allows recommended annotation is personalized and varies from user to user.
- Third, our system is interactively user adaptive, i.e. the more a user annotates his/her photos via our system, the more accurate the recommended annotations are.

The rest of this paper is organized as follows. In section II, we review the background and related words in image retrieval and image classification. The core steps of the BoW model and how we conduct experiments are presented in section III. Section IV shows experiment results and evaluations. The conclusion and future works are discussed in section V.

II. Background & Related works

There are many approaches to build an Image Information Retrieval System. Some methods aim at high precision, i.e achieve high quality of top retrieved results, while others focus on high recall, i.e retrieve all positive results. Among them, the first effective and scalable method is Bag-of-Words, Sivic and Zisserman [9], which is inspired by the correspondence algorithm using in text retrieval. Before going into details of BoW model in subsection II-B, we will first introduce some different methods for image retrieval problem in subsection II-A.

A. Different approaches for image retrieval problem

One of many popular methods is histogram comparisons which compares 2 different images based on their color histograms. Some early works of this approach using a cross-bin matching cost for histogram comparison can be found in [12], [13], [14]. In [14], Peleg et al. represent images as sets of pebbles after normalization. The similarity score is then computed as the matching cost of two sets of pebbles based on their distances.

Another well-known technique is template matching, i.e. seeking a given pattern in a image by comparing to candidate regions of the same size in the target image. By consider both the pattern and candidate regions as a length-N vector, we can compare these two vectors by using different kinds of distance metrics, one such metric is the Minkowski distance [15]. The major disadvantage of 2 listed methods is that they require the query and target images to share a similar stationary interrelation, which means that components of the given image are not allowed to change freely in a certain extent. Bag-of-Words, the method that is discussed in this paper, is another approach that can tolerate the flexibility in structures of the object and thus, have a wider variation of application in many problems.

B. Bag-of-Words

Since Bag-of-Words is originally a text retrieval algorithm, we will first introduce some backgrounds about BoW in text retrieval problem in subsection II-B1 before discussing using BoW in image retrieval in subsection II-B2.

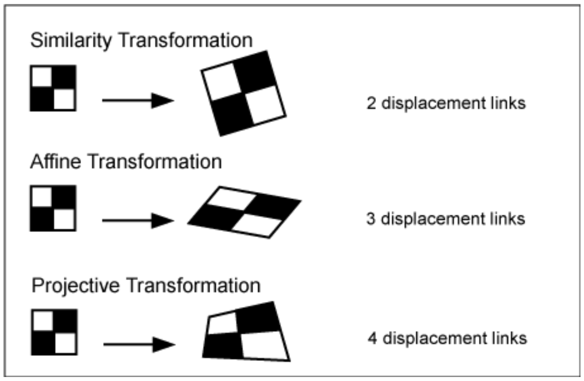


Fig. 1: Transformation methods (Source: geodata.ethz.ch)

1) Bag-of-Words in text retrieval: In text retrieval, a text is represented as a histogram of words, also known as BoW [16]. This scheme is called term frequency weighting as the value of each histogram bin is equal to the number of times the word appears in the document. Moreover, some words are less informative than others since those words appear in almost every document. Therefore, we need a weighting scheme that address this problem. Such weighting scheme is called inverse document frequency (idf) and is formulated as $log(N_D/N_i)$, where N_D is the number of documents in the collection and N_i is the number of documents which contains word i. The overall BoW representation is thus weighted by multiplying the term frequency (tf) with the inverse document frequency (idf) giving rise to the tf-idf weighting [16]. In addtion, extremely frequent words, “stop words”, can be removed entirely in order to reduce storage requirements and query time.

2) Bag-of-Words in image retrieval: When applying BoW to image retrieval, a major obstacle is the fact that text documents are naturally broken into words by spaces, dots, hyphens, or commas. In contrast, there is no such separator in images. Therefore, the concept of “visual word” is introduced where each visual word is represented as a cluster obtained using k-means on the local descriptor vectors [9].

The bigger the vocabulary size is, the more different the visual words are. Hence, the vocabulary helps us distinguish the images more effectively. Nonetheless, with bigger vocabulary size, slightly different descriptors can be assigned to different visual words thus not contributing to the similarity of the respective images and causing a drop in performance examined in [17], [18], [11]. Philbin et al. [11] suggests “soft assign” method where each descriptor is assigned to multiple nearest visual words instead of using “hard assignment”, i.e only assign a local descriptor to only one nearest visual word. Despite its effectiveness, this method also significantly costs more storage and time.

C. Geometric transformation in images

In computer, an image usually represents as a matrix of pixels. In other words, it is an array I which I(x, y) is value of the pixel at x and y in horizontal and vertical coordinates, respectively. A geometric transformation T is a rule or formula which transform every pair of (x, y) into (x', y'), i.e. $(x', y') = T(x, y)$. Based on how different a transformed image is, compared to the original one, geometric transformation is divided into many types (e.g. similarity, affine, projective, polynomial ...). In this section, the authors describe some geometric terminologies related to our work.

1) Similarity transformation: A similarity transformation is a transformation matrix T such that a new image A' is obtained by applying T into all pairs of (x, y) in A, i.e. $A'(x', y') = A(x, y)$. The main property of similarity transformation is that A and A' have the same shape. In other words, corresponding sides in A and A' are in proportion, corresponding angles in A and A' have the same measure. Similarity transformation includes scaling, translation, rotation, reflection and compositions of them in any combination and order.

2) Affine transformation: Similar to similarity transformation, an affine transformation is also a matrix T that transform an image A into a new image A'. However, angles between lines as well as distances between points may not be preserved, though ratios of distances between points lying on a straight line are preserved. Affine transformation includes similarity transformation, homothety, shear mapping and compositions of them in any combination and order.

3) Projective transformation: Generalized from affine transformation, projective transformation may not preserve the parallelism, i.e. two parallel lines/planes can be transformed into two intersecting lines/planes.

Because of the difference in complexity among transformations, the estimation process of different transformations requires different number of pairs (x, y) and (x', y'). More specifically, affine and projective transformations require at least 2, 3 and 4 pairs of coordinates, respectively.

D. Evaluation the performance of image retrieval system

Since there are many different algorithms lying behind different information retrieval systems, it is crucial to have a measurement for evaluating the performance of information retrieval systems. In this subsection, we will describe different popular measurements which are used to estimate performance of an information retrieval system. First of all, the authors would like to introduce 3 things that are required to evaluate information retrieval systems: a document collection, a set of queries, and a set of relevant documents for each query.

The 2 early born and fundamental measures are precision and recall. While precision is calculated as the ratio between the number of relevant documents that are retrieved and the total number of retrieved documents, recall is the quotient between the number of relevant documents retrieved and the number of relevant documents. The formula of these 2 measures are shown below:

$$recall = \frac{number\ of\ relevant\ items\ retrieved}{number\ of\ relevant\ items\ in\ collection} \quad (1)$$

$$precision = \frac{number\ of\ relevant\ items\ retrieved}{total\ number\ of\ items\ retrieved} \quad (2)$$

Despite their simplicity, precision and recall fail to take into account of the order of retrieved documents which is also important since a user want to find his/her desired document as fast as possible. Therefore, many other measures, which also consider the arrangement of the retrieved documents, are developed based on precision and recall. There is one such measure named Average Precision (AP), the average value of the precision value achieved from the top documents cut-off at positions where relevant documents are retrieved. In addition to precision, recall, and AP, there are also many other measures such as F-measure, i.e. weighted average of precision and recall or R-precision, i.e. the precision at R-th position where R is the number of relevant documents.

In our experiments, to evaluate the result, the authors choose to use the AP measure along with the mean value of AP overall queries, namely mean Average Precision (mAP). Besides the fact that these 2 measures can fully evaluate both the content and the order of the ranked list, their popularity are also a reason why the authors choose them. AP and mAP are also used in many previous works in image retrieval so we can easily compare our framework's performance with others based on these 2 measures.

III. Method

A. Overview of an Image Retrieval System

As described in Figure 2, a typical Image Retrieval System consists of 2 main steps: Feature Extraction and Similarity Matching. The Feature Extraction step is explained in detail in section III-B. Subsequently, in section III-C, the authors focus on describing how we perform Similarity Matching with our implementation of BoW model.

B. Feature Extraction

To detect and extract features from images, there are many methods that have been proposed (Harris-Affine, Hessian-Affine detectors [19], Maximally stable extremal region (MSER) detector [20], Edge-based region detector [21], Intensity extrema-based region detector [22] ...). The authors choose to use Hessian-Affine detector, for detecting and extracting features from images. By using Hessian-Affine detector, which is also used in other baseline methods, our experiment's result can easily be compared with other ones.

As we tested on Oxford Building 5K Dataset [23], there are typically 3,300 features for each image and a total about 16 millions of features for the whole dataset. Then, we compute the SIFT descriptor [24] of all the features and these descriptors is used for matching images in the next step.

C. Similarity Matching

As described in Figure 3, our proposed framework would consist of the following major parts: Dictionary Building, Quantization, tf-idf Weighting and Spatial Rerank. The 3 former steps are described in section III-C1, section III-C2, and section III-C3. The last step is our main focus in this paper and is discussed in section III-C4.

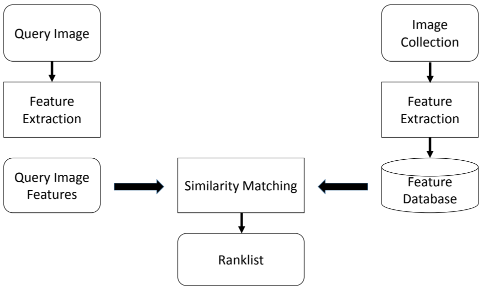


Fig. 2: How an Image Retrieval System works

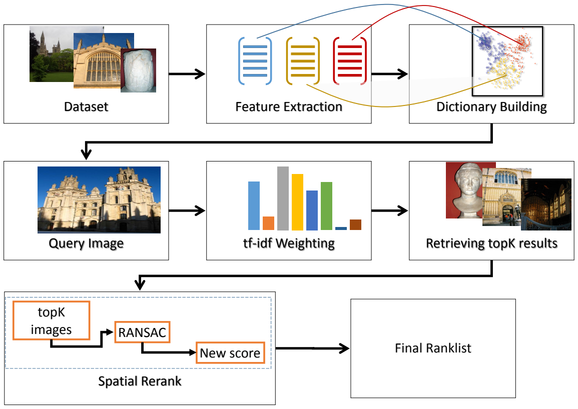


Fig. 3: Proposed framework

1) Dictionary Building: Treating each descriptor as an individual visual words in the dictionary results in a worthless waste of resources and time. In order to overcome this obstacle, the authors therefore build the dictionary by considering some similar descriptors as one. In other words, all descriptor vectors are divided into k clusters, each representing a visual word. There are many algorithms that are proposed to solve this kind of problem. However, the authors use the approximate k-means (AKM). AKM is proposed by Philbin et al. [10]. Comparing to the original k-means, AKM can reduce the majority amount of time taken by exact nearest neighbors computation but only gives slightly different result. Also, in [10], Philbin et al. shows that using 1M dictionary size would have the best performance on the Oxford Building 5K Dataset [23].

2) Quantization: Subsequently, each 128-dimension SIFT descriptor needs to be mapped into the dictionary. Commonly, each descriptor is assigned into the nearest word in the dictionary. Thus, when two descriptors are assigned to diferent words, they are considered as totally different. In practice, this hard assignment leads to errors due to variability in descriptor (e.g. image noise, varying scene illumination, instability in the feature detection process ...) [11]. In order to handling this problem, the authors use soft assignment instead of hard assignment. In particular, each 128-dimension SIFT descriptor is reduced to a k-dimension vector of their k nearest visual words in the dictionary. Each of these k nearest cluster is assigned with weights calculated from the formula proposed by Sivic et al. [11], $weight = \exp(-\frac{d^2}{2\delta^2})$, where d is the distance from the cluster center to descriptor point. Then, by adding all these weights to their corresponding bins, we will have the BoW representation of an image.

Fig. 4: Different images recorded from the same object under various viewpoints

In this work, k and δ^2 are chosen to be 3 and 6250, respectively.

3) tf-idf Weighting Scheme: As mentioned in section II, tf-idf is a popular weighting scheme that is used by almost any BoW model. In this section, the authors will show how this scheme is applied to our system.

For a term t_i in a particular document d_j, its term frequency $tf_{i,j}$ is defined as follow:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{3}$$

Where $n_{i,j}$ is the number of occurrences of the considered term t_i in the document d_j. The denominator is the sum of the number of occurrences of all the terms in document d_j.

The inverse document frequency idf_i of a term t_i is computed by the following formula:

$$idf_i = \log \frac{|D|}{|\{j : t_i \in d_j\}|} \tag{4}$$

Where, $|D|$ is the total number of documents in the corpus, $|\{j : t_i \in d_j\}|$ is the number of documents where the term t_i appears, i.e. $n_{i,j} \neq 0$

The tf-idf weight of a term t_i in a document d_j is then calculated as the product of tf and idf:

$$tfidf_{i,j} = tfi,j \times idf_i \tag{5}$$

The tf-idf weight is then used to compute the similarity score between an image d_i and a query q:

$$s_{d_i,q} = \vec{tfidf}_i \cdot \vec{tfidf}_q = \sum_{j=1}^{|T|} tfidf_{i,j} \times tfidf_{q,j} \tag{6}$$

Finally, by sorting the list of images corresponding to their similarity score with a query, we achieve the raw ranked list of this query which is then used for the Spatial Rerank step.

4) Spatial Rerank: When applying BoW model into documents, we often ignore the spatial structure of words. However, the spatial structures of words in documents, especially images, are important for retrieving and ranking. Therefore, we push the spatial information of visual words into the original BoW model by incorporating the spatial constrains to the top ranked images and rerank them.

The spatial verification process evaluates a geometric transformation of a image and the query. In detail, we need to estimate the geometric transformation matrix that transforms features of the query to features of a image. As described in Figure 4, objects can be taken under various viewpoints and it means the parallelism may not be preserved. Consequently, a projective transformation , which requires at least a set of 4 matched pairs of points to be estimated, is needed in the transformation matrix. A common approach is to use RANSAC [25], i.e. choosing 4 matched pairs of points randomly to generate different transformation hypotheses multiple times and taking the hypotheses which has the largest number of inliers.

The spatial verification process evaluates the geometric transformation between the query image and each image in the top-k ranked results. More specifically, given a query and an image, we need to estimate the geometric transformation matrix that transforms features of the query to features of the image. As shown in Figure 4, images of objects can be taken under various viewpoints and it means the parallelism may not be preserved. As a result, the computation of the projective transformation matrix II-C requires a set of at least 4 matched pairs of feature points between the query and the image (a feature x in the query and a feature y in the image are called a matched pair if they belong to the same cluster of visual word). Additionally, the measurement of the transformation must also be taken into account, since we need to find a transformation that can accurately satisfies the spatial relation. Thus, the authors consider the number of inliers to be the measurement for a transformation. A matched pair (x, y) is an inlier if apply the computed matrix on coordinate of feature x would produce x' such that x' and y are approximately in the same position. One common approach to find the needed transformation matrix is to apply RANSAC algorithm [25], i.e. repeatedly choosing 4 matched pairs of points randomly to generate different transformation hypotheses, the hypothesis which has the highest measurement (in other words, the largest number of inliers) is the geometric transformation that we need. Finally, with this chosen geometric transformation, we then add the idf weight of the inliers to the new similarity score and rerank these top-k ranked images. In figure 5, we show an example image of RANSAC algorithm.

In our system, the authors choose the number of iterations for the RANSAC algorithm is 100 times and perform spatial rerank on the top 800 retrieved result of the dataset, which is showed to obtain the best accuracy by Philbin et al. [10].

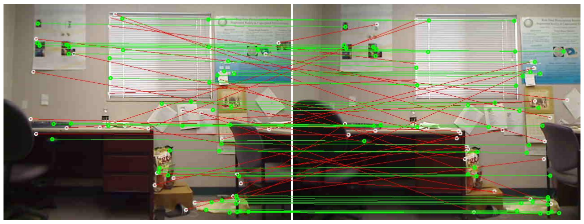


Fig. 5: Example of RANSAC algorithm. Verified matched features are colored green, unverified matched features are colored red

IV. Experiment & Result

Oxford5K. To prove our hypotheses, the authors test both the BoW systems with and without spatial rerank on the Oxford Building 5K Dataset [23]. This dataset was constructed by Philbin et al. in 2007 [10]. It consists of 5,062 images of resolution 1024×768 belongs to 11 different Oxford buildings. Images for each building are collected from Flickr by searching using text queries. In figure 6, some samples from the dataset are shown. Along with the dataset, there are also 55 queries along with their ground-truth, 5 for each landmark, as shown in figure 7. The groundtruth of 55 queries are manually constructed. For each query, images are classified into 4 groups: (1) *Good*: the building appears apparently, (2) *OK*: more than 25% of the building is present, (3) *Bad*: the building is not shown up, and (4) *Junk*: less than 25% of the building is captured. The reason why the authors use this dataset is because of its popularity, it is used by many previous works in this field. Thus, we can easily compare our systems with those previous works.

Evaluation protocol. The performance of our system is evaluated by mean average precision (mAP). Since mAP is only described briefly in II, we now give detailed formulas of mAP. If the set of relevant documents for q query $q_j \in Q$ is $\{d_1, ...d_mj\}$ and R_jk is the set of ranked retrieval results from the top result until you get to document d_k, mAP(Q), Q is the set of queries, is defined as follow:

$$mAP(Q) = \frac{1}{|Q|} \times \sum_{j=1}^{|Q|} \frac{1}{m_j} \times \sum_{k=1}^{m_j} Precision(R_{j_k}) \qquad (7)$$

To ensure the objectivity of evaluation process, the authors decide to use the already implemented C++ code to calculating mAP of the ranklist available at [23]. In this implementation, since AP approximates the area under the precision-recall curve of a query, mAP is therefore computed as the average area under precision-recall curves of all queries. Additionally, the set of relevant images for a query is defined as those images which are categorized as *Good* or *OK* in the corresponding groundtruth.

Our experiment shows that spatial rerank has significant impact on the retrieval quality of BoW model, an increase from 0.676 to 0.741 in term of mAP. The original BoW model have AP at least or higher than 0.5 for 41/55 queries

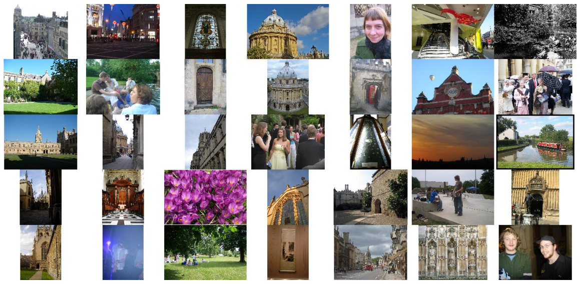


Fig. 6: Some random images from Oxford Building 5K Dataset

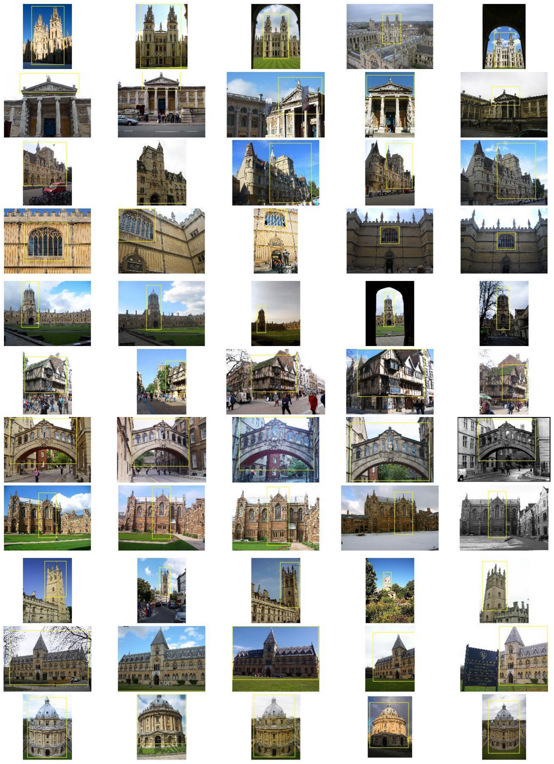


Fig. 7: 55 queries of Oxford Building 5K Dataset

and by using spatial rerank the figure is improved to 47/55 queries. Among 55 queries, there are 40 queries that achieve higher AP after incorporating spatial information and there 22 queries increasing more than 0.050. The highest boost is 0.583, from 0.417 to 1.000. However, there 2/55 queries suffering significant performance drop (decreases more than 0.100). To explain why there are performance reduction in these 2 queries, shown in figure 8, the authors believe that they are affected by the background features which actually should not be considered in the rerank step. For better illustration, the APs of all 55 queries are given in 9.

V. Conclusion

Through our experiments, it is proved that spatial rerank significantly boosts the performance of BoW model. This is a very potential result to further improve the performance of

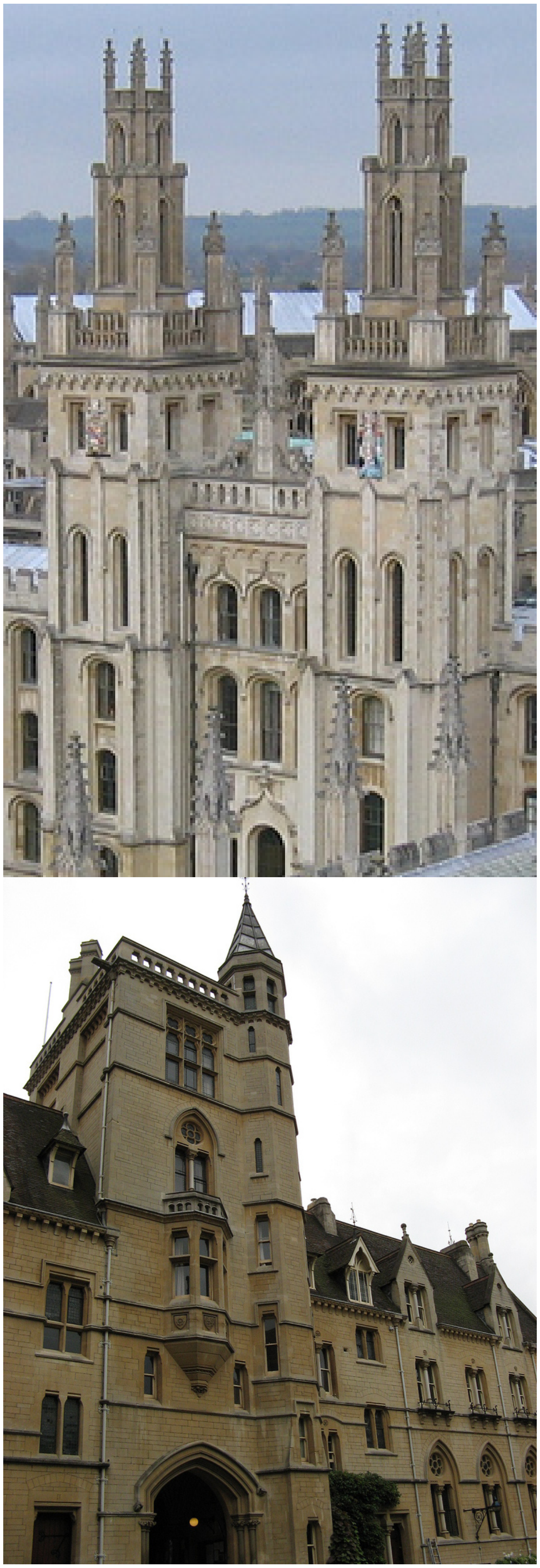
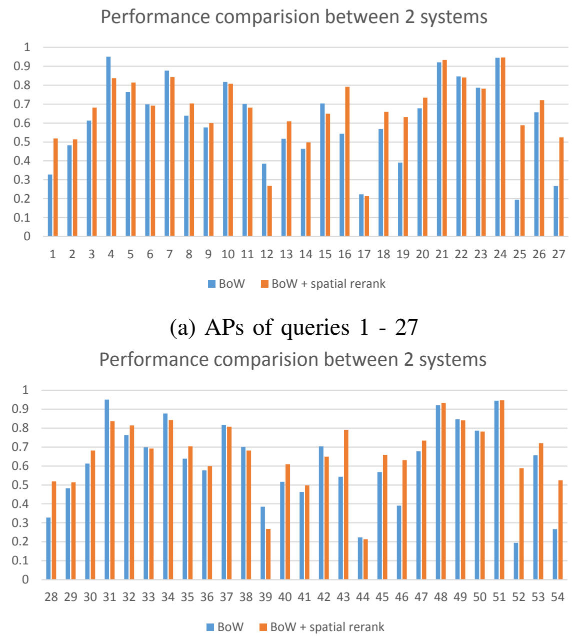


Fig. 8: Two queries having worse performance after using spatial rerank



(a) APs of queries 1 - 27

(b) APs of queries 28 - 55

Fig. 9: Comparison chart between the 2 methods

many image retrieval systems. In the future, we plan to keep upgrading our system to operate on other datasets which are larger in size and also more variant in term of content. Our final goal is to deploy our system for real-time usage with limited computer resources.

References

[1] A. M. Kaplan and M. Haenlein, "Users of the world, unite! the challenges and opportunities of social media," *Business Horizons*, vol. 53, no. 1, pp. 59 – 68, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0007681309001232

[2] R. E. Wilson, S. D. Gosling, and L. T. Graham, "A review of facebook research in the social sciences," *Perspectives on Psychological Science*, vol. 7, no. 3, pp. 203–220, May 2012.

[3] R. Brunelli, *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley, 2009.

[4] A. Rosenfeld and G. J. Vanderburg, "Coarse-fine template matching," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 7, no. 2, pp. 104–107, Feb 1977.

[5] M. Gharavi-Alkhansari, "A fast globally optimal algorithm for template matching using low-resolution pruning," *Image Processing, IEEE Transactions on*, vol. 10, no. 4, pp. 526–533, Apr 2001.

[6] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and texture-based image segmentation using em and its application to content-based image retrieval," in *Computer Vision, 1998. Sixth International Conference on*, Jan 1998, pp. 675–682.

[7] Y. Rubner, C. Tomasi, and L. Guibas, "The earth mover's distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, 2001, pp. I–511–I–518 vol.1.

[9] J. Sivic and A. Zisserman, "Video google: A text retrieval approach to object matching in videos," in *Proc. ICCV*, 2003.

[10] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. CVPR*, 2007.

[11] J. Philbin, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proc. CVPR*, 2008.

[12] H. C. Shen and A. K. Wong, "Generalized texture representation and metric," *Computer Vision, Graphics, and Image Processing*, vol. 23, no. 2, pp. 187 – 206, 1983. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0734189X83901123

[13] M. Werman, S. Peleg, and A. Rosenfeld, "A distance metric for multidimensional histograms," *Computer Vision, Graphics, and Image Processing*, vol. 32, no. 3, pp. 328 – 336, 1985. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0734189X85900556

[14] S. Peleg, M. Werman, and H. Rom, "A unified approach to the change of resolution: space and gray-level," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 11, no. 7, pp. 739–742, Jul 1989.

[15] W. Ouyang, F. Tombari, S. Mattoccia, L. Di Stefano, and W.-K. Cham, "Performance evaluation of full search equivalent pattern matching algorithms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 1, pp. 127–143, Jan 2012.

[16] C. D. Manning, P. Raghavan, and H. Schtze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[17] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. CVPR*, 2006.

[18] G. Schindler, M. Brown, , and R. Szeliski, "City-scale location recognition," in *Proc. CVPR*, 2007.

[19] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, 2004.

[20] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions." in *BMVC*, P. L. Rosin and A. D. Marshall, Eds. British Machine Vision Association, 2002.

[21] T. Tuytelaars and L. V. Gool, "Content-based image retrieval based on local affinely invariant regions," in *In Int. Conf. on Visual Information Systems*, 1999, pp. 493–500.

[22] ——, "Wide baseline stereo matching based on local, affinely invariant regions," in *In Proc. BMVC*, 2000, pp. 412–425.

[23] [Online]. Available: http://www.robots.ox.ac.uk/ vgg/data/oxbuildings/

[24] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981. [Online]. Available: http://doi.acm.org/10.1145/358669.358692