BOOSTING OBJECT RETRIEVAL BY ESTIMATING PSEUDO-OBJECTS

Kuan-Hung Lin[1], *Kuan-Ting Chen*[1], *Winston H. Hsu*[1], *Chun-Jen Lee*[2], *Tien-Hsu Li*[2]

National Taiwan University[1], Chunghwa Telecom Co., Ltd.[2], Taipei, Taiwan

ABSTRACT

State-of-the-art object retrieval systems are mostly based on the bag-of-visual-words representation which encodes local appearance information of an image in a feature vector. A search is performed by comparing query object's feature vector with those for database images. However, a database image vector generally carries mixed information of an entire image which may contain multiple objects and background. Search quality is degraded by such noisy (or diluted) feature vectors. We address this issue by introducing the concept of pseudo-objects to approximate candidate objects in database images. A pseudo-object is a subset of proximate feature points in an image with its own feature vector to represent a local area. We investigate effective methods (e.g., Grid, G-means, and GMM-BIC) to estimate pseudo-objects. Experimenting over two consumer photo benchmarks, we demonstrate the proposed methods significantly outperforming other state-of-the-art object retrieval algorithms.

Index Terms**—**image retrieval, object retrieval, pseudo-object, visual word

1. INTRODUCTION

Object-based image retrieval is to find the occurrences of specific objects in the image/photo database. This is different from the typical problem of searching for database images that globally resemble the query image. In an object retrieval system, images containing the query object are retrieved even though they may globally look quite different from the query image (i.e., retrieving photos containing "Starbucks" logos). In other words, the system aims to discover high similarity between local regions of images instead of full images. In practice, the user provides a query image and marks the region that properly contains the query object. The system then returns a ranked list of images that are likely to contain the query object. Figure 1 shows an example of object query and corresponding search results.

Prior successful object retrieval systems, such as [9][7][2], are based on vector space framework adopted from text retrieval domain with visual words taking the place of textual words. Visual words have the advantage of being effective at capturing salient local object characteristics in images while being invariant to changes in image capturing conditions, such as variations in scale, viewing angle, or lighting. The construction of visual words involves image feature point detection, description and quantization. Feature points are detected by feature point detectors [6] (cf. Figure 2) and then described by Scale-Invariant Feature Transform (SIFT) [5]. The descriptors from training images are quantized into clusters. The centroid of each cluster is then defined as a visual word. A feature point is assigned to its nearest visual word in high dimensional descriptor space. Therefore, every image can be viewed as a collection of discrete visual words (bag-of-visual-words) and empirically represented by a frequency histogram of visual words. Similarity between images is computed from normalized frequency histograms which can be viewed as feature vectors. The vector space model also has the benefit that feature vectors can be computed and indexed off-line to speed up the on-line query process.

This approach however has its shortcomings. Although feature points encode local appearance information, the histogram representation ignores vital spatial information and compares images globally. While this may be acceptable in textual document retrieval, it is a significant drawback for object retrieval because the feature points of an object must assemble in a unique way, usually within close proximity of each other. In [4], Lazebnik et al. proposed a spatial pyramid matching scheme to improve upon the bag-of-visual-words model. In their approach, images are repetitively divided into finer grids to form a spatial pyramid. Visual word histograms are computed for each grid at each level and corresponding histograms between images are compared for similarity. This is to ensure that images are deemed similar when they not only look alike globally, but also have corresponding sub-components that resemble each other.

The aforementioned approaches are both constrained in object retrieval by the fact that retrieval is carried out by comparing the query region, which may occupy only a small proportion of the query image, to the content of full database images. They are only applicable to images where the query object takes up a major proportion of the entire image or stands in front of a relatively non-noisy background. At the same time, images that contain the query object along with many other objects are not retrieved because their visual word histogram (enhanced with spatial pyramid or not) are diluted by visual words of other objects or background in the entire image.

Intuitively, it is best if locations of objects inside images can be known beforehand. When a query object is given, direct object to object search is performed instead of object to full image search that is the current norm. The computation of object histograms is a two-fold problem. Firstly, automatic segmentation of objects must be performed to identify objects in images. This is still an open research problem in the computer vision community. Secondly, each feature point needs to be assigned to segmented objects if available through manual annotation. However, it is not always possible to determine the membership of a feature point especially when it lies near object boundaries. Besides, manual annotation for modern image database comprised of billions of images is formidable.

To boost the emerging object retrieval, we propose matching over the *pseudo-object* – a subset of proximate feature points in an image with its own feature vector to represent a local area (cf. Figure 2). Instead of tackling the extremely hard problem of computing (or manually annotating) true objects, we take an approximating approach where feature points are clustered according to their spatial coordinates. It is natural since salient objects are generally characterized by those proximate feature points. Each cluster is assumed to be a pseudo-object and its histogram is computed and normalized to obtain a feature vector. With pseudo-objects, in an automatic manner, we aim to better represent candidate objects in the image database by additional feature vectors. This unavoidably results in more storage space required and query processing speed will also take a hit. These problems can be mitigated by the use of modern multi-core and multi-cluster parallel computing platforms [1]. Furthermore, our approach is fully compatible with and can further improve other techniques (e.g., spatial verification, query expansion [2], and reranking [10]), in the promising visual word paradigm.

In summary, the main contributions of our work include:

- Identifying the deficiencies in modern object retrieval systems based on object to full image comparison.
- Proposing pseudo-object representation that averts the problem mentioned above and investigating effective methods (e.g., Grid, G-means and GMM-BIC) for estimating pseudo-objects.
- Demonstrating the significant effectiveness of the proposed methods by experimenting over two large-scale consumer photo benchmarks.
- The proposed approach is fully compatible with and can benefit existing retrieval techniques (i.e., spatial verification, query expansion, reranking, etc.).

2. PROPOSED METHODS

We propose three methods for estimating the pseudo-objects by clustering the proximate feature points, characterizing the salient regions in the images. The first method is Grid inspired by the spatial pyramid scheme [4]. The second method is the G-means algorithm [3] which is an improved version of *k*-means that automatically determines the number of groups (i.e., clusters), *k*. The third method is GMM-BIC [8], a Gaussian mixture model with Bayesian information criterion to determine the suitable number of Gaussian components. The latter two methods are able to estimate the number of groups and are desirable since the best number of candidate objects in an image cannot be known. In addition, these methods are not restricted by grid boundaries. Better representation of objects that span over grids is possible. Figure 1 demonstrates the effectiveness of the proposed methods by showing the top 4 retrieval results of the query in Figure 2-(a).

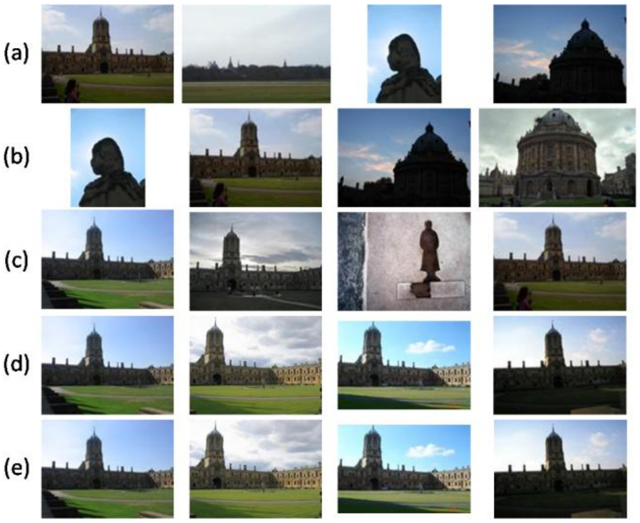


Figure 1. Retrieval results of the 'Christ_church_3' query in the Oxford Buildings Dataset [7]. The query object is marked by a yellow rectangle in **Figure 2**-(a). The rows show top 4 results of (a) query object matching to global image histograms [2][7][9], (b) spatial pyramid matching [4], matching over *pseudo-objects* estimated by (c) Grid, (d) G-means, and (e) GMM-BIC. The last two methods, more flexible in estimating the number and location of pseudo-objects, clearly outperform the others.

The estimation of pseudo-objects is a clustering problem on the 2-dimentional coordinates of feature points in an image. The goal is to form clusters that approximate objects in the image. An ideal cluster would be one whose feature points are all from the same candidate object; visual word histogram computed using feature points in this cluster is then a good representation of the object. In the following sections we describe three different methods to obtain feature point clusters (also illustrated in Figure 2).

2.1. Grid

An image is divided evenly along each dimension, forming four grids of the same size. An extra grid of the same size is placed at the center of the image assuming that it is where people tend to place important objects. Each grid is considered as a pseudo-object with corresponding feature vector. With this method, every image is considered as having five pseudo-objects in it.

2.2. G-means

The G-means algorithm [3] works with the assumption that data points in a cluster follows a Gaussian distribution. This translates to an object having feature points distributed spherically around its center. The further away a feature point is from the object center, the less likely it belongs to that object. The algorithm starts by running *k*-means with a small *k*. Each of the resulting cluster centers is statistically tested to detect whether its clustered points are sampled from a Gaussian. A center is retained if so or else split into two new centers. The set of retained and new centers are then used as initial centers for the next *k*-means routine. This process repeats until no more cluster splits. Note that the cluster number is determined automatically.

2.3. GMM-BIC

With the assumption that feature points of a pseudo-object should follow a Gaussian distribution, we attempt to fit a Gaussian Mixture Model (GMM) to the feature points in an image. Again, the number of pseudo-objects (or Gaussian components) is unknown. Therefore, we adopt the Bayesian Information Criterion (BIC) [8], which penalizes model complexity in determining the optimal number of Gaussians. This process iterates a few times to obtain a list of GMMs with different number of Gaussians. We then choose the GMM with minimized BIC and use posterior probabilities to assign feature points to different pseudo-objects.

3. EXPERIMENTS

We conduct object retrieval experiments on two datasets collected from *Flickr*, the popular photo-sharing website.

3.1. Data Sets

Oxford Buildings Dataset: The Oxford Buildings Dataset [7] consists of 5062 images collected by issuing Oxford landmark names as search keywords on the Flickr website. The images are downsized to quarter of the original to match the image dimensions in our next dataset. The average image size is about 512x374 pixels.

Flickr11k dataset: The Flickr11k dataset is a subset taken from the Flickr550 dataset [10]. We modify the queries and ground truth defined by the authors to suit our (content-based) object query criteria. The result is a total of 1,282 ground truth images in 7 query categories with 8 queries each. We then add 10,000 images random sampled from Flickr550 to form our Flickr11k dataset. The average image size is about 500x360 pixels.

3.2. Implementation and Evaluation

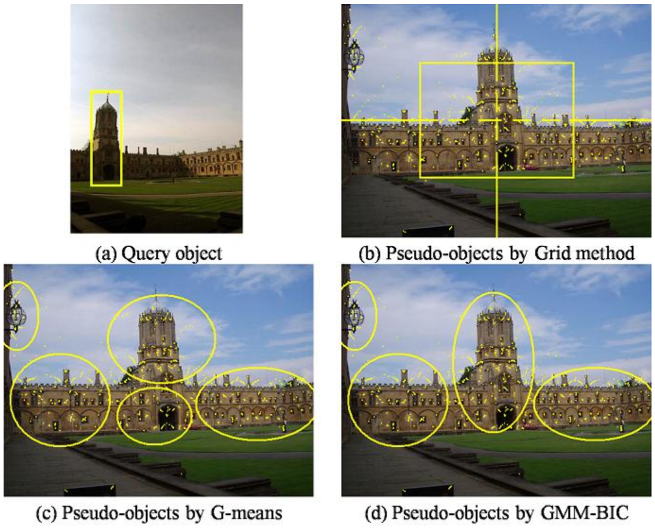
We use Hessian-affine detector [6] to extract feature points in each image. The feature points are described by SIFT [5] and quantized into 3500 visual words. Every image or pseudo-object is thus represented by a 3500-dim. feature vector, a normalized visual word histogram. The query vector constructed according to query object region is compared to the feature vectors in database using L1 distance. Similarity score of a database image, *I*, is defined as:

$$score(I) = \max_{v_d \epsilon V_I} \left(-L1(v_q, v_d)\right),$$

where v_q is the query vector, v_d is the feature vector of a pseudo-object, and V_I is the set of feature vectors of pseudo-objects in the image *I*. Note that we are using MAX (maximum) for fusing similarity scores for the query to all pseudo-objects in the image. Other methods (e.g., AVG, MIN, etc.) had been preliminarily experimented but yielded poor performances.

For each query, the system returns a list of database images ranked by their scores in descending order. A PR-curve is plotted and average precision (AP) is defined as the area under the curve. Category performance is evaluated by averaging the APs of queries in the same category. Overall system performance is evaluated by mean average precision (mAP) which is the mean of all query APs in the dataset.

The baseline is that V_I contains only one feature vector that represents the entire image [9][2][7]. Spatial pyramid [4] up to level 1 is adopted here for comparison. For the G-means and GMM-BIC methods, though adaptive, an upper-bound of 5 pseudo-objects for each image is set.


(a) Query object

(b) Pseudo-objects by Grid method

(c) Pseudo-objects by G-means

(d) Pseudo-objects by GMM-BIC

Figure 2. (a) A sample query objet marked by a yellow rectangle. Pseudo-objects, illustrated by yellow circles, by different methods: (b) Grid; when grid lines pass through true objects, pseudo-objects become less effective; (c) G-means and (d) GMM-BIC are free of restrictions by grid lines and can better approximate (true) objects. Note that the dots represent the locations of detected feature points.

3.3. Discussion

Table 1 summarizes query performance by categories in the Oxford Buildings Dataset over the baseline method – one visual word vector only for the entire image. Generally, G-means and GMM-BIC gain the most improvements over the baseline and spatial-pyramid methods. The two proposed pseudo-object estimation methods can catch local

Table 1. Search performance on the Oxford Buildings Dataset. The right column for each method shows percentage change in performance over the baseline method [2][7][9]. Categories that benefit significantly from our methods are highlighted in bold fonts. Their average gain is shown at the bottom line of each dataset. The "MAP selected categories" (last row) shows the MAP over categories benefiting from pseudo-objects. Note that the detailed performance over Flickr11k, exhibiting similar behaviors as that in the Oxford Buildings Dataset, is not shown here due to space limitation.

Oxford Buildings Dataset	Baseline	Spatial pyramid [4]		Grid		G-means		GMM-BIC	
All_souls	0.439	0.450	2.5%	0.461	5.1%	0.460	4.8%	0.453	3.1%
Ashmolean	**0.228**	**0.219**	**-3.9%**	**0.307**	**35.1%**	**0.290**	**27.4%**	**0.321**	**40.9%**
Balliol	0.305	0.316	3.8%	0.305	0.0%	0.305	0.0%	0.305	0.0%
Bodleian	**0.061**	**0.070**	**16.3%**	**0.113**	**86.1%**	**0.117**	**93.7%**	**0.120**	**99.0%**
Christ_church	**0.201**	**0.201**	**0.1%**	**0.254**	**26.7%**	**0.293**	**46.0%**	**0.286**	**42.4%**
Cornmarket	0.404	0.420	4.0%	0.412	2.1%	0.404	0.0%	0.416	3.0%
Hertford	0.442	0.458	3.7%	0.442	0.0%	0.442	0.0%	0.442	0.0%
Keble	**0.284**	**0.291**	**2.7%**	**0.343**	**20.8%**	**0.347**	**22.2%**	**0.395**	**39.3%**
Magdalen	**0.051**	**0.051**	**-1.0%**	**0.069**	**34.6%**	**0.067**	**31.0%**	**0.065**	**26.7%**
Pitt_rivers	0.512	0.537	5.0%	0.512	0.0%	0.512	0.0%	0.512	0.0%
Radcliffe_camera	0.556	0.563	1.3%	0.556	0.0%	0.556	0.0%	0.556	0.0%
MAP overall	0.316	0.325	2.8%	0.343	8.4%	0.345	8.9%	0.352	11.2%
MAP selected categories	**0.165**	**0.166**	**1.0%**	**0.217**	**31.8%**	**0.223**	**35.3%**	**0.237**	**44.1%**

characteristics and respectively improve up to 35.3% and 44.1% (the average of categories benefitting from pseudo-objects) over the baseline method. The improvement is especially obvious when the candidate objects in database images are relatively small compared to the entire image; e.g., in categories such as 'Bodleian,' 'Ashmolean,' 'Christ_church,' etc. Note that the two methods are adaptive and require no pre-fixed cluster (or pseudo-object) number.

The spatial pyramid [4] only enhances query categories whose target objects usually appear at the center of ground truth images, leaving little space for background noises and interferences from other objects. However, if the set of ground truth images exhibits enough intra-variation as query object appearing in different sizes and at different locations, the spatial pyramid method degrades the performance (cf. Table 1). The Grid method requires the least computing power and achieves satisfactorily. However, it is inevitable that some objects are divided into multiple grids and results in less informative feature vectors and waste of storage space. As the image database grows more diverse, this problem may become more serious (cf. Figure 2).

The G-means method improves less than GMM-BIC in the Oxford Buildings Dataset. It is because the underlying *k*-means algorithm tends to generate more circular clusters. For example, tower-shaped objects are often broken down into smaller round sections. The GMM-BIC method generates clusters that better match true objects.

The proposed pseudo-object methods work by adding additional feature vectors into the database. The original full image feature vector is left unaffected – assumed as another pseudo-object. Therefore, performance loss is unlikely to happen even if object to full image comparison is more suitable for a query category (selected by the MAX fusion).

4. CONCLUSION

We propose novel pseudo-object for improving object level retrieval in consumer photos and investigate automatic methods (e.g., Grid, G-means, GMM-BIC) for estimating pseudo-objects. We showed that the proposed methods boost search performance significantly by experimenting on two large benchmarks. The GMM-BIC is shown effective for pseudo-object estimation and significantly outperforms other state-of-the-art object retrieval algorithms.

In the future, we plan to investigate means to mitigate the possible side effects of a large number of additional feature vectors. A possibility is to utilize powerful multi-core multi-computer platforms such as the Google cluster architecture [1]. We will also study promising techniques for dimension reduction applied to our framework with extra pseudo-object features.

5. REFERENCES

[1] L.B. Barroso, et al., "Web search for a planet: the Google cluster architecture," IEEE *Mirco*, vol. 23, March-April 2003.

[2] O. Chum, et al., "Total recall: automatic query expansion with a generative feature model for object retrieval," in *Proceedings of ICCV*, 2007.

[3] G. Hamerly and C. Elkan, "Learning the k in k-means," in *Proceedings of NIPS*, 2003.

[4] S. Lazebnik, C. Schmid and J. Ponce, "Beyond bags of features: spatial pyramid matching for recognizing natural scene categories," in *Proceedings of CVPR*, 2006.

[5] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91-110, November 2004.

[6] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *IJCV*, vol. 60, no. 1, October 2004.

[7] J. Philbin, et al., "Object retrieval with large vocabularies and fast spatial matching," in *Proceedings of CVPR*, 2007.

[8] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461-464, March 1978.

[9] J. Sivic and A. Zisserman, "Video Google: a text retrieval approach to object matching in videos," in *Proceedings of ICCV*, 2003, vol. 2, pp. 1470-1477.

[10] Y.-H. Yang, et al., "ContextSeer: context search and recommendation at query time for shared consumer photos," in *Proceedings of ACM Multimedia*, 2008.