Image Re-Ranking Using Query-Specific Semantic Signature with Result Analysis

[1]Ms. Tejashree Kumar Shinde, [2]Prof. Prakash. B. Dhainje and [3]Dr.Deshmukh Pradeep K
[1]Student of M.E (CSE), [2]MTech,MBA(IT),PhD(CSE), Head of CSE Department and
[3]M.E.,PhD(CSE), Principal,
[1,2,3]Shriram Institute of Engineering & Technology, Solapur University, Solapur

Abstract- *For searching images, Image Search engines mostly use keywords and they rely on surrounding text. Indistinctness of query images is hard to describe accurately by using keywords. Eg. If Apple is query keyword then categories can be "apple laptop" , "red apple", etc. Without online training low level features may not well co-relate with high level semantic meaning is one challenge. Some Low-level features are sometimes incompatible with visual observation. To get semantic signature the visual and textual features of images are then projected into their related semantic spaces. In online stage images are re-ranked by comparing semantic signature obtained from semantic space obtained from query keywords. By just 20 – 30 concepts Semantic space of a query keyword can be described these are referred as "reference classes".*

Keywords: *K-Means Algorithm, Semantic Signatures, Canny Edge Detection, Re-ranking, framework*

I. INTRODUCTION

To search images web-scale image search engines mostly uses keywords as queries and rely on surrounding text to search images. So they suffer from the indistinctness of query keywords. For example, using "apple" as query, the retrieved images belong to different category, such as "apple logo", and "apple laptop" "red apple". An effective way to improve the image search results is Online image re-ranking [5, 4, 9].The re-ranking strategy is adopted by Major internet image search engines [5]. Its diagram is shown in Figure 1. After giving a query keyword input by a user, according to a stored word-image index file, a pool of images appropriate to the query keyword are retrieved by the search engine. A user is going to select a query image, which reflects the user's search meaning, from the pool, the remaining images in the pool are re-ranked based on their visual similarity with the query image. The visual appearances of images are pre-computed offline and stored by the search engine. The online computational cost of image re-ranking is on comparing visual features. In order to achieve high effectiveness, the visual feature vectors need to be short and their matching needs to be fast. One of the major challenge is that the similarities of low level visual features may not well correlate with images' high-level semantic meanings which understand users' search intention. To narrow down this semantic gap, for offline image detection and retrieval, there have been a number of study to map visual features to a set of predefined concepts or attributes as semantic signature [11, 7, 15]. However, these approaches are only applicable to closed image sets of quite small sizes. They are not appropriate for online web- based image re-ranking.

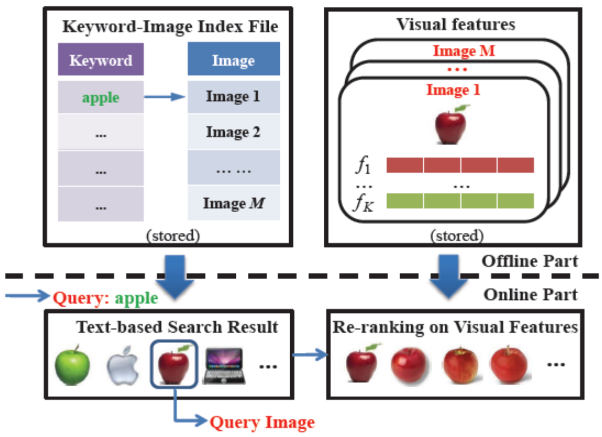


FIGURE 1. THE USUAL IMAGE RE-RANKING FRAMEWORK.

A. Approach

In our paper, a novel framework is projected for web image re-ranking. It learns different visual semantic spaces for different query keywords individually and automatically Instead of constructing a universal concept dictionary. We suppose that the semantic space related to the images to be re-ranked can be considerably narrowed down by the query keyword provided by the user. For example, the semantic concepts of "mountains" and "Paris" are unlikely to be relevant and can be ignored if the query keyword is "apple". The query-specific visual semantic spaces can more accurately model the images to be reranked, since they have removed other potentially unlimited number of irrelevant concepts, which serve only as noise and weaken the performance of re-ranking in terms of both accuracy and computational cost. The visual features of images are then predictable into their related visual semantic spaces to get semantic signatures. Images are re-ranked by comparing their semantic signatures obtained at the online stage.

B. Related Work

For calculation of image similarity content-based image retrieval uses visual features. To learn visual similarity metrics to capture users search intent this Relevance feedback [13, 16, 14] was widely used. It required more users effort to selection of multiple relevant and irrelevant image examples and often needs online training. Cui et al. [5, 4] estimated an image re-ranking approach which limited users effort to just one-click feedback. Simple image re-ranking approach has been adopted by popular web-scale image search engines such as Bing and Google recently.

The main component of image re-ranking is to compute the visual similarities between images. Nowadays many image features[8, 6, 2, 10] have been developed. For different query images, low-level visual features which are effective for one image category may not work well for another to address this, Cui et al. [5, 4] categorized the query images into eight predefined intention categories and gave different feature weighting schemes to different types of query images. It was not easy for only eight weighting schemes to cover the large diversity of all the web images. It was also possible for a query image to be classified to a wrong category.

In recent times, for general image recognition and matching there have been a number of works on using predefined concepts or attributes as image signature. To address this Rasiwasia et al. [11] mapped visual features to a universal concept dictionary and Lampert et al. [7] used predefined qualities with semantic meanings to detect novel object classes. Few approaches [1, 15, 12] transferred knowledge between object classes by measuring the similarities between novel object classes and known object classes (known as reference classes). All these reference-classes were universally applied to all the images and their training data was manually selected. They are more fit for offline databases with lower diversity (such as animal databases [7, 12] and face databases [15]) such that object classes better share similarities. For modeling all the web images, a huge set of concepts or reference classes are required, which is impractical and unsuccessful for online image re-ranking.

II. APPROACH OVERVIEW

The diagram of our approach is shown in Figure 2. In the offline stage, the reference classes, which represent different semantic concepts of query keywords are automatically discovered. For a query keyword e.g. "apple" a set of most relevant keyword expansions such as "red apple", "apple macbook", and "apple iphone" are automatically selected considering both textual and visual information. This keyword development set defines the reference classes for the query keyword. For repeatedly obtain the training examples of a reference class, the keyword expansion e.g. "red apple" is used to retrieve images by the search engine. Images retrieved by the keyword expansion "red apple" are much less varied than those retrieved by the original keyword "apple". After repeatedly removing outliers, the retrieved top images are used as the training examples of the reference class. Few reference classes such as "apple laptop" and "apple macbool" have similar semantic meanings and their training sets are visually similar. To improve the efficiency of online image re-ranking, unnecessary reference classes are removed. For each query keyword, a multi-class classifier on low level visual features is trained from the training sets of its reference classes and stored offline. If there are K types of visual features, one could merge them to train a single classifier. It is possible to train a separate classifier for each type of features. Our experiments show that the latter choice can increase the re-ranking accuracy but will also increase storage and lessen the online matching efficiency because of the increased size of semantic signatures.

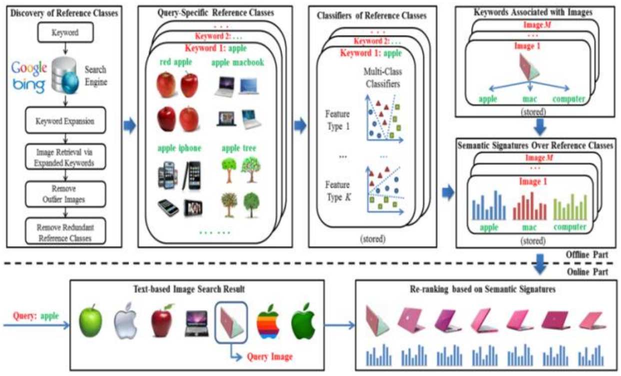


Figure 2 - Diagram Of Our New Image Re-Ranking Framework

An image may be appropriate to multiple query keywords. So it could have several semantic signatures obtained in different semantic spaces. According to the word image index file, each image in the database is associated with a few relevant keywords. For each significant keyword, a semantic signature of the image is extracted by calculating the visual similarities between the image and the reference classes of the keyword using the classifiers trained in the preceding step. The reference classes are the basis of the semantic space of the keyword. At offline stage If an image has N relevant keywords, then it has N semantic signatures to be computed and stored . At the online stage, according to the query keyword input by a user pools of images are retrieved by the search engine. As all the images in the pool are relevant to the query keyword, they all have pre-computed semantic signatures in the semantic space of the query keyword. Once the user chooses a query image, all the images are re-ranked by comparing similarities of the semantic signatures.

III. METHODOLOGY

A. Algorithm

1. There are two parts online and offline parts.

2. At online stage reference classes representing different concepts related to query keywords are automatically exposed. For a query keyword e.g. “apple” a set of most relevant keyword expansions such as “red apple” and “apple macbook”are automatically selected utilizing both textual and visual information.

3. Set of keyword Expansions describe reference classes for different keywords.

4. A multi class classifier is qualified on training set of reference classes.

5. If there are k types of visual and textual features like color, shape, texture we can combine them to train single classifier.

6. At online stage pool of images are retrieved according to query keyword. One time user chooses query image semantic signatures are used to compute similarities of image with pre-computed semantic signatures.

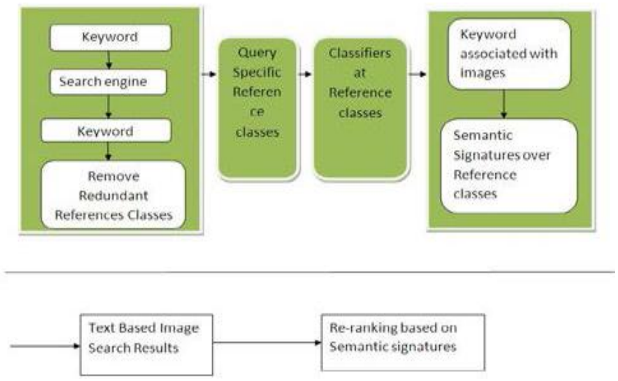


Figure 3.Semantic Approach Of Re-Ranking Of Images

B. Kmeans Algorithm For Clustering Of Images

K-Means Algorithm Properties

•There are always K clusters. [19]
•There is always at least one item in each cluster.
•The clusters are non-hierarchical and they do not overlap.

•Every member of a cluster is closer to its cluster than any other cluster because nearness does not always involve the 'center' of clusters.

The K-Means Algorithm Process

•The dataset is partitioned into K clusters and the data points are randomly assigned to the clusters resulting in clusters that have roughly the same number of data points.

•For each data point:

• Calculate the distance from the data point to each cluster.

• If the data point is closest to its own cluster, leave it where it is. If the data point is not closest to its own cluster, move it into the closest cluster.

•Repeat the above step until a complete pass through all the data points results in no data point moving from one cluster to another. At this point the clusters are stable and the clustering process ends.

•The choice of initial partition can greatly affect the final clusters that result, in terms of inter-cluster and intra cluster distances and cohesion.

K-means algorithm

1) Select K points for initial group centroids.
2) Each object is assigned to the group that has the closest distance to the centroid.
3) After all objects have been assigned, recalculate the positions of the K centroids.
4) Steps 2 and 3 are repeated until the centroids no longer move.

This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

C. Duplication Image Detection

A duplicate image discovery system generates an image table that maps hash codes of images to their corresponding images. The image table may group images according to their group identifiers generated from the most important elements of hash codes based on significance of elements representing an image. The image table thus segregate images by their group identifiers. To detect a duplicate image of a target image, the discovery system generates a target hash code for target image. The discovery system then selects the images associated with those similar hash codes as being duplicates of the target image. Duplicate detection done during image upload phase.

D. Keyword Expansion

For a keyword q, automatically define its reference classes through finding a set of keyword expansions E(q) most applicable to q. A set of images S(q) are retrieved by the search engine using q as query based on textual information. Keyword expansions are create from the words extracted from the images in S(q)3. A keyword expansion Eq is expected to frequently appear in S(q).For reference classes to well confine the visual content of images, we require that there is a subset of images which all contain e and have similar visual content.

Based on these considerations, keyword expansion are found in a search-and-rank way as follows.

For each image I 2 S(q), all the images in S(q) are reranked according to their visual similarities (defined in [5]) to I. The T most frequent words $W_I = \{ W_I^1, W_I^2 \ldots\ldots W_I^T \}$ among top D re-ranked images are found. If a word w is among the top ranked image, it has a ranking score rI (w) according to its ranking order; otherwise rI (w) = 0,

$$rI(w) = \begin{cases} T - j \; (1) & w = w_I^j \\ 0 & w \notin W_I \end{cases} \qquad (1)$$

The overall score of a word w is its accumulated ranking scores over all the images,

$$r(w) = \sum_{I \in S} rI(w). \qquad (2)$$

The P words with highest scores are selected and combined with the original keyword q to form keyword expansions, which define the reference classes. In this experiment, T = 3, D = 16, and P = 30.

E. Training Images of Reference Classes

Each keyword expansion e is used to retrieve images from the search engine and top K images are kept. Since the keyword expansion e has less semantic ambiguity than the original keyword q, the images retrieved by e are much less diverse than those retrieved by q. Once removing outliers by k-means clustering, these images are used as the training examples of the reference class.

F. Redundant Reference Classes

To compute comparison between two reference classes, we use half of the data in both classes to train a SVM classifier to classify the other half data of the two classes. If they can be easily separated, then the two classes are considered not similar. Suppose n reference classes are obtained from the previous steps. The training images of reference class i are split into two sets, $A_i^{1 \text{ and }} A_i^2$. In order to measure the distinctness D(i; j) between two reference classes i and j, a two-class SVM is trained from A_i^1 and A_j^1 for each image in A_i^2 the SVM classifier output a score indicating its probability of belonging to class i. Assume the averaging score over A_i^2 is p_i Similarly, the averaging score pj over A_i^2 is also computed. Then D(i; j) = h((pi + pj)/2), where h is a monotonically increasing function. In our approach, it is defined as

$$h(\overline{p}) = 1 - e^{-\beta(\overline{p} - \alpha)} \quad (3)$$

where β and α are two constants. When (pi + _pj) /2 goes below the threshold α, h(p) decreases very quickly so as to

penalize pair-wisely similar reference classes. We empirically select α = 0.6 and β = 30.

G. Reference Class Selection

We finally select a set of reference classes from the n candidates. The keyword expansions of the selected reference classes are most revelant to the query keyword q. The relevance is defined by Eq (2) in Section D Meanwhile, we require that the selected reference classes are not similar with each other such that they are diverse enough to characterize different aspects of its keyword. The distinctiveness is measured by the n x n matrix D defined in Section F. The two criterions are simultaneously satisfied by solving the following optimization problem.

We introduce an indicator vector $y \in \{0,1\}^n$ such that $y_i = 1$ indicates reference class i is selected and $y_i = 0$ indicates it is removed. y is estimated by solving,

$$\arg \max_{y \in \{0,1\}^n} \{ \lambda R_y + y^T D_y \} \quad (4)$$

Let e_i be the keyword expansion of reference class i. R =($r(e_1)$,…, $r(e_n)$), where $r(e_i)$ is defined in Eq (2). λ is the scaling factor used to modulate the two criterions. Since integer quadratic programming is NP hard, we relax y to be in R^n and select reference classes i whose yi $\geq$ 0.5.

SEMANTIC SIGNATURE

If given M reference classes for keyword q and their training images automatically retrieved, a multi-class classifier on the visual features of images is trained and it outputs an M-dimensional vector p, representing the probabilities of a new image I belonging to different reference classes. Then p is used as semantic signature of I. The distance between two images Ia and Ib are measured as the L1-distance between their semantic signatures pa and pb,

$$d(Ia; Ib) = \|p^a - p^b\|_1$$

A Combined Features vs Separate Features

For train the SVM classifier, we adopt six types of visual features used in [5], attention guided color signature, color spatialet, wavelet, multi-layer rotation invarient edge orientation histogram, histogram of gradients, and GIST. They distinguish images from different perspectives of color, shape, and texture. The joint features have around 1; 700 dimensions in total. A natural idea is to merge all types of visual features to train a single powerful SVM classifier which better differentiate different reference classes. The purpose of using semantic signatures is to capture the visual content of an image, which may belong to none of the reference classes, instead of classifying it into one of the reference classes. If there are N types of independent visual features, it is actually more effective to train separate SVM classifiers on different types of features and to combine the N semantic signatures $\{p^n\}^N_{n=1}$ from the outputs of N classifiers. The N semantic signatures explain the visual content of an image from different aspects e.g. color, texture, and shape and can better differentiate images outside the reference classes. For example, in following Figure 4, "red apple" and "apple tree" are two reference classes. A new image of "green apple" can be well characterized by two semantic signatures from two classifiers taught on color features and shape features separately, since "green apple" is similar to "red apple" in shape and similar to "apple tree" in color. Then the distance between two images Ia and Ib is,

$$d(Ia; Ib) = \sum_{n=1}^{N} w_n \|p^{a,n} - p^{b,n}\|_1$$

where w_n is the weight on different semantic signatures and it is specified by the query image Ia selected by the user. Wn is decided by the entropy of pa;n,

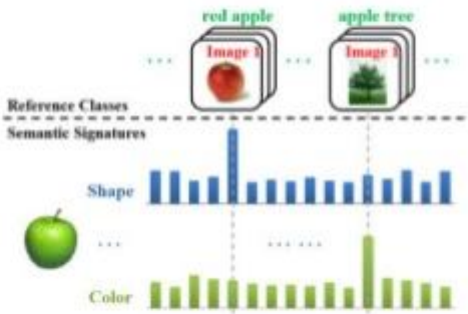


Figure 4. Describe "Green Apple" Using Reference Classes. Its Shape Is Captured By Shape Classifier Of "Red Apple" And Its Color Is Captured By Color Classifier Of "Apple Tree".

Wn = 1 / 1+eH(pa;n)

H(pa;n) =- $\sum_{i=1}^{M} P_i^{a,n} \ln P_i^{a,n}$

If pa;n consistently distributes over reference classes, the nth type of visual features of the query image cannot be well characterized by any of the reference classes and we assign a low weight to this semantic signature.

a. B. Locality sensitive hashing technique for nearest neighbor search

The proposed system works as follows:

1. There are two parts: offline part and online part as shown in above Fig.
2. Firstly, a user has to submit a text query for searching images. This text will be taken as a query keyword by the search engine.
3. Then at the offline stage, Keyword Expansion is done to accurately capture the user's search intension by considering the words frequently co-occurring with the query keyword and synonyms and meaning of query keyword. These keyword expansions will be taken as reference classes of the query keyword.
4. Then images of expanded keywords will be retrieved.
5. After that, a user has to select one query image. And at the offline stage, the visual query expansion is done automatically just by one click on query image to get multiple positive example images specific to the query image to accurately users' intention.
6. The new image re-ranking framework focuses on the semantic signatures associated with the images derived using a trained multiclass classifier. The semantic signatures of the query image and visually expanded images are acquired by comparing their visual features with the reference classes of the query keyword using this trained multiclass classifier.
7. Also the semantic signatures of the remaining images in the image set are derived in similar manner in the same semantic space of the query keyword.
8. These semantic signatures are further reduced by using LSH hashing techniques to further increase their matching efficiency. The study says that Perceptual hash is reliable and fastest algorithm for web-based applications.
9. As all the images in the image set have pre-computed hash values. So at the online stage, the images in this set are re-ranked by comparing their hash values, using Euclidean Distance formula to compute image similarities with the query image.
10. And these finally re-ranked images are displayed to user

Algorithm:

1. **Hashing Algorithm**

An *LSH family* $\mathcal{F}$ is defined for a metric space $\mathcal{M} = (M, d)$, a threshold $R > 0$ and an approximation factor $c > 1$. This family $\mathcal{F}$ is a family of functions $h : \mathcal{M} \to S$ which map elements from the metric space to a bucket $s \in S$. The LSH family satisfies the following conditions for any two points $p, q \in \mathcal{M}$, using a function $h \in \mathcal{F}$ which is chosen uniformly at random:

- if $d(p,q) \leq R$, then $h(p) = h(q)$ (i.e., p and q collide) with probability at least P_1,
- if $d(p,q) \geq cR$, then $h(p) = h(q)$ with probability at most P_2.

A family is interesting when $P_1 > P_2$. Such a family $\mathcal{F}$ is called (R, cR, P_1, P_2)*-sensitive*.

Alternatively[4] it is defined with respect to a universe of items U that have a similarity function $\phi : U \times U \rightarrow [0, 1]$. An LSH scheme is a family of hash functions H coupled with a probability distribution D over the functions such that a function $h \in H$ chosen according to D satisfies the property that

$$Pr_{h \in H}[h(a) = h(b)] = \phi(a, b)$$

for any $a, b \in U$.

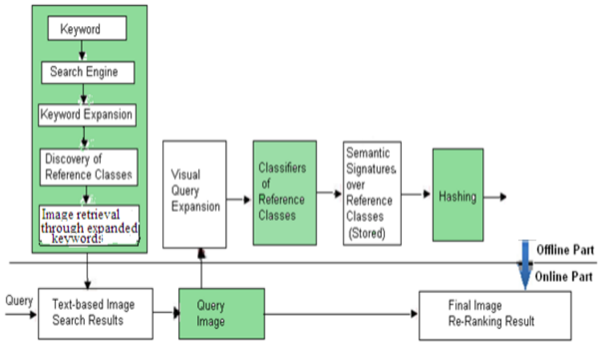
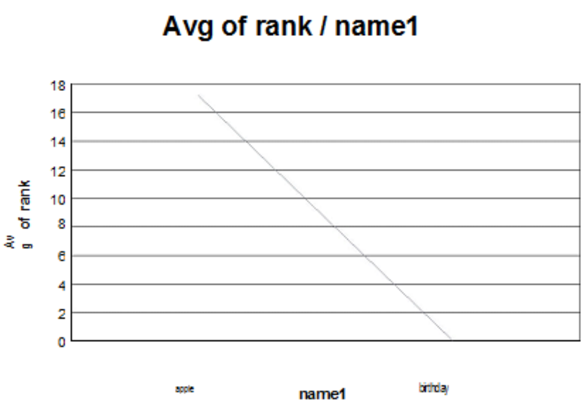
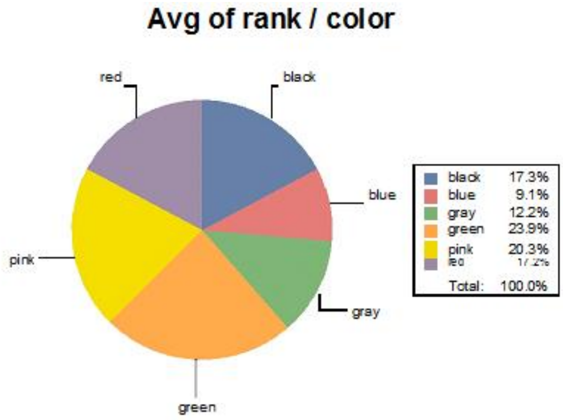


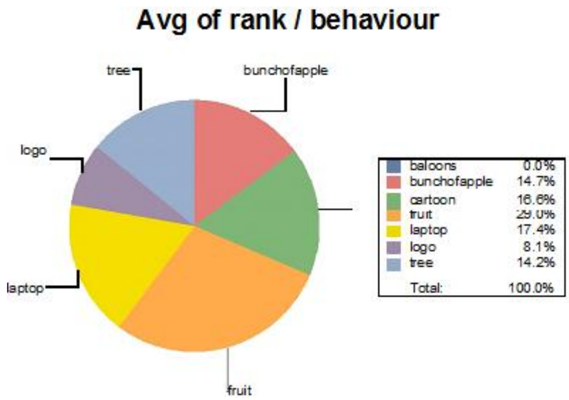
Figure 5: Locality Sensitive Hashing Technique For Nearest Neighbor Search

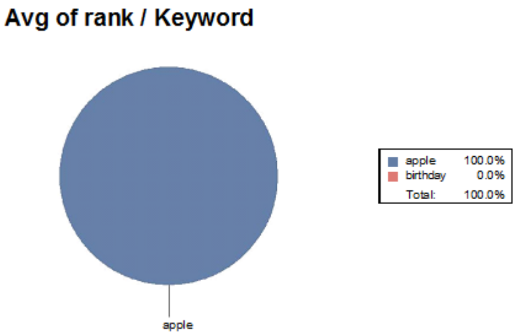
IV. EXPERIMENTAL RESULTS

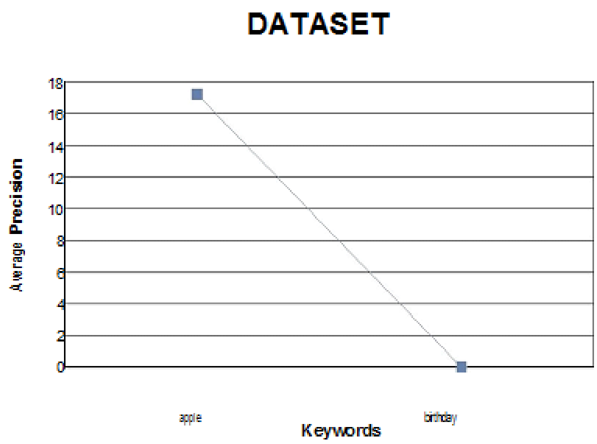
The images for testing the performance of re-ranking and the images of reference classes can be collected at different time and from different search engines. Given a query keyword, 1000 images are retrieved from the whole web using certain search engine. We create three data sets to evaluate the performance of our approach in different scenarios. In data set I, one of the feature ie color is used in second dataset II behavior of image is used and dataset III keyword is used for ranking. In the result the graph is shown according to the ranking of that images two keywords apple and birthday are used.

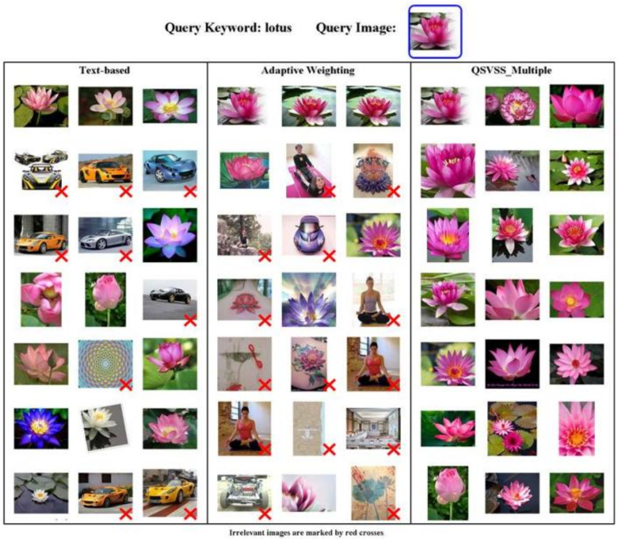












V. RERANKING PRECISIONS

Averaged top m accuracy is used as the evaluation criterion. Top m accuracy is defined as the proportion of relevant images among top m re-ranked images. Applicable images are those in the same category as the query image. Averaged top m accuracy is obtained by averaging top m precision for every query image not including outlier. We take on

this criterion as an alternative of the precision-recall curve since in image re-ranking, the users are most concerned about the qualities of top retrieved images instead of number of relevant images returned in the whole result set. We compare with two benchmark image re-ranking approaches used in [5]. They directly compare visual features. (1) Global Weighting ie Predefined fixed weights are adopted to combine the distances of different low-level visual features. (2) Adaptive Weighting [5] ie planned adaptive weights for query images to fuse the distances of different low-level visual features. It is adopted by Bing Image Search. For our new approaches, two different ways of computing semantic signatures as discussed in Section IV.A are compared.

Query-specific visual semantic space using single signatures (QSVSS Single). For an image a single semantic signature is computed from one SVM classifier trained by combine all types of visual features.

Query-specific visual semantic space using multiple signatures (QSVSS Multiple).To an image, multiple semantic signatures are calculated from multiple SVM classifiers, each of which is trained on one type of visual features separately.

Some parameters used in our approach are tuned in a small separate data set and they are fixed in all the experiments. Our approach significantly out performs Global Weighting and Adaptive Weighting, which directly compare visual features. On data set I, our approach enhances the averaged top 10 precision from 44:41% (Adaptive Weighting) to 55:12% (QSVSS Multiple). 24:1% relative improvement has been achieved. In our approach, computing multiple semantic signatures from separate visual features has higher precisions than computing a single semantic signature from combined features. However, it costs more online computation since the dimensionality of multiple semantic signatures is higher. if the testing images for re-ranking and images of reference classes are collected from different search engines, the performance is slightly lower than the case when they are collected from the same search engine. However, it is still much higher than directly comparing visual features. This indicates that we can utilize images from various sources to learn query-specific semantic spaces. even if the testing images and images of reference classes are collected at different times relevant months apart query specific semantic spaces still can effectively improve re-ranking. Compared with Adaptive Weighting, the averaged top 10 precision has been improved by 6:6% and the averaged top 100 precision has been improved by 9:3%. This indicates that once the query-specific semantic spaces are learned, they can remain effective for a long time and do not have to be updated frequently.

A. Reranking

Images outside the reference classes It is interesting to know whether the learned queryspecific semantic spaces are effective for query images which are outside the reference classes. To answer this question, if the category of an query image corresponds to a reference class, we deliberately delete this reference class and use the remaining reference classes to train SVM classifiers and to compute semantic signatures when comparing this query image with other images. We repeat this for every image and calculate the average top m precisions. This evaluation is denoted as RmCategoryRef and is done on data set III6. Multiple semantic signatures QSVSS Multiple are used. The results are shown in Figure 5. It still greatly outperforms the approaches of directly comparing visual features. This result can be explained from two aspects. (1) The multiple semantic signatures obtained from different types of visual features separately have the capability to characterize the visual content of images outside the reference classes. (2) Many negative examples images belonging to different categories than the query image are well modeled by the reference classes and are therefore pushed backward on the ranking list.

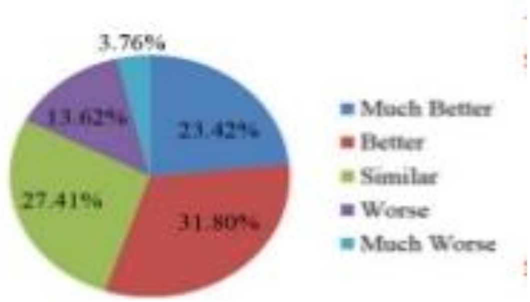
B. Query specific semantic space vs. universal semantic space

In previous works [11, 7, 1, 15, 12], a universal set of reference classes or concepts were used to map visual features

to a semantic space for object recognition or image retrieval on closed databases. In our experiment, we assess whether this approach is applicable to web-based image re-ranking and compare it with our approach. We randomly select M reference classes from the whole set of reference classes of all the 120 query keywords in data set I. The M selected reference classes are used to train a universal semantic space in a way similar to Section 4.1. Multiple semantic signatures are obtained from different types of features separately. This universal semantic space is applied to data set III for image re-ranking.. M is chosen as 25, 80, 120 and 160[7]. This method is denoted as UnivMClasses. When the universal semantic space chooses the same number 25of reference classes as our query-specific semantic spaces, its precisions are no better than visual features. Its precisions increase when a larger number of reference classes are selected, the gain increases very slowly when M is larger than 80. Its best precision when M = 160 are much lower than QSVSS Multiple and even lower than RmCategory Ref, even though the length of its semantic signatures is five times larger than ours.

C. User study

User experience is significant for web-based image search. In order to fully reflect the extent of users' approval, user study is conducted to compare the results of our approach QSVSS Multiple compared with Adaptive Weighting on data set I. Twenty users are invited. Eight of them are familiar with image search and the other twelve are not. To avoid bias on the evaluation, we ensure that all the participants do not have any knowledge about the current approaches for image re-ranking, and they are not told which results are from which methods. Each user is assigned 20 queries and is asked to randomly select 30 images per query. Each selected image is used as a query image and the re-ranking results of Adaptive Weighting and our approach are shown to the user. The user is required to indicate whether our re-ranking result is "Much Better", "Better", "Similar", "Worse", or "Much Worse" than that of Adaptive Weighting. 12; 000 user comparison results are collected. The comparison results are shown in Figure 6. In over 55% cases our approach delivers better results than Adaptive Weighting and only in less than 18% cases ours is worse, which are often the noisy cases with few images relevant to the query image exists.



CONCLUSIONS

We propose a new image re-ranking framework, which learns query-specific semantic spaces to considerably improve the competence and efficiency of online image reranking. The visual features of images are projected into their related visual semantic spaces automatically learned through keyword expansions at the offline stage. The extracted semantic signatures can be 70 times shorter than the original visual feature on average, while achieve 20%-35% relative improvement on re-ranking precisions over state-of the- art methods.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to Prof. for their supervision and guidance. We would like to express our appreciation to our parents and all the teachers and lecturers who help us to understand the importance of knowledge and show us the best way to gain it.

REFERENCES

[1] E. Bart and S. Ullman. Single-example learning of novel classes using representation by similarity. In Proc. BMVC, 2005.

[2] Y. Cao, C. Wang, Z. Li, L. Zhang, and L. Zhang. Spatial-bag-offeatures. In Proc. CVPR, 2010.

[3] G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. In Proc. NIPS, 2001.

[4] J. Cui, F. Wen, and X. Tang. Intentsearch: Interactive on-line image search re-ranking. In Proc. ACM Multimedia. ACM, 2008.
[5] J. Cui, F. Wen, and X. Tang. Real time google and live image search re-ranking. In Proc. ACM Multimedia, 2008.
[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Proc. CVPR, 2005.
[7] C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In Proc. CVPR, 2005.
[8] D. Lowe. Distinctive image features from scale-invariant keypoints. Int'l Journal of Computer Vision, 2004.
[9] B. Luo, X. Wang, and X. Tang. A world wide web based image search engine using text and image content features. In Proceedings of the SPIE Electronic Imaging, 2003.
[10] J. Philbin, M. Isard, J. Sivic, and A. Zisserman. Descriptor Learning for Efficient Retrieval. In Proc. ECCV, 2010.
[11] N. Rasiwasia, P. J. Moreno, and N. Vasconcelos. Bridging the gap: Query by semantic example. IEEE Trans. on Multimedia, 2007.
[12] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele What helps wherevand why? semantic relatedness for knowledge transfer. In Proc. CVPR, 2010.
[13] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: a power tool for interactive content-based image retrieval. IEEE Trans. on Circuits and Systems for Video Technology, 1998.
[14] D. Tao, X. Tang, X. Li, and X. Wu. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2006.
[15] Q. Yin, X. Tang, and J. Sun. An associate-predict model for face recognition. In Proc. CVPR, 2011.
[16] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. Multimedia Systems, 2003.