Annotating Images by Mining Image Search Results

Xin-Jing Wang, Lei Zhang, *Member*, *IEEE*, Xirong Li, and Wei-Ying Ma, *Senior Member*, *IEEE*

Abstract—Although it has been studied for years by the computer vision and machine learning communities, image annotation is still far from practical. In this paper, we propose a novel attempt at model-free image annotation, which is a data-driven approach that annotates images by mining their search results. Some 2.4 million images with their surrounding text are collected from a few photo forums to support this approach. The entire process is formulated in a divide-and-conquer framework where a query keyword is provided along with the uncaptioned image to improve both the effectiveness and efficiency. This is helpful when the collected data set is not dense everywhere. In this sense, our approach contains three steps: 1) the search process to discover visually and semantically similar search results, 2) the mining process to identify salient terms from textual descriptions of the search results, and 3) the annotation rejection process to filter out noisy terms yielded by Step 2. To ensure real-time annotation, two key techniques are leveraged—one is to map the high-dimensional image visual features into hash codes, the other is to implement it as a distributed system, of which the search and mining processes are provided as Web services. As a typical result, the entire process finishes in less than 1 second. Since no training data set is required, our approach enables annotating with unlimited vocabulary and is highly scalable and robust to outliers. Experimental results on both real Web images and a benchmark image data set show the effectiveness and efficiency of the proposed algorithm. It is also worth noting that, although the entire approach is illustrated within the divide-and-conquer framework, a query keyword is not crucial to our current implementation. We provide experimental results to prove this.

Index Terms—Clustering, information filtering, object recognition, real-time systems.

1 Introduction

The number of digital images has exploded with the advent of digital cameras, which requires effective image search techniques. Although it is an intuitive way of conducting image search since "a picture is worth a thousand words," the Query-By-Example (QBE) scheme (i.e., using images as queries) is seldom adopted by current commercial image search engines. The reasons are at least twofold: 1) The semantic gap problem, which is also the fundamental problem in the Content-based Image Retrieval (CBIR) field. This is because current visual feature extraction techniques are not effective enough in representing the semantics of an image. 2) Computational expensiveness. It is well known that the inverted indexing technique ensures the practical usage of current text search engines, which store the keyword-document relationship in a so-called inverted index file whose entries are keywords and whose values are the documents. The information about which document contains a certain keyword can thus be obtained in real time. Given a textual query, the search results are the intersection of the documents indexed by the query keywords individually (if no ranking functions applied). However, since images are 2D media and the spatial relationship between pixels is crucial in conveying the semantics of an image, how to define image "keywords" is still an open question. This prevents the inverted indexing technique from being directly applied to image search, which results in a critical efficiency problem for QBE retrieval.

Due to these reasons, there has been a surge of interest in image autoannotation and object recognition in recent years. Researchers have tried to define ways to automatically assign keywords onto images or image regions and proposed many learning models [1], [2], [4], [9], [15], [17], [21], [28], [32], [36]. A big problem they encountered is the lack of training data. It is known that to manually label images is very expensive [35] and it tends to produce inconsistent annotations on which many questions should be addressed beforehand: Which kind of images can be labeled consistently? How do we define the strategy to ensure consistency, etc.?

Differently from the previous approaches which adopt Computer Vision or Machine Learning techniques, we investigate how effective a data-driven and model-free approach which leverages commented-upon images on the Web could be.

Imagine that a large-scale image set is available so that, for each query image, at least one duplicate can be detected; what we need to do then is just to annotate the query with the duplicate's textual descriptions. A valuable resource from which we can collect such a data set is the Web; it not only contains large numbers of images, but these images generally have human-assigned comments or descriptions.

However, it is too idealistic to require, for each image, a well-annotated duplicate, but we can leverage a group of

- *X.-J. Wang, L. Zhang, and W.-Y. Ma are with Microsoft Research Asia, 4F Sigma Center, 49 Zhichun Road, Haidan District, Beijing 100190, P.R. China. E-mail: {xjwang, leizhang, wyma}@microsoft.com*
- *X. Li is with the Intelligent Systems Lab Amsterdam, University of Amsterdam, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands. E-mail: xirong@science.uva.nl.*

Manuscript received 23 Sept. 2007; revised 10 Feb. 2008; accepted 5 May 2008; published online 14 May 2008.
Recommended for acceptance by J.Z. Wang, D. Geman, J. Luo, and R.M. Gray. For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMISI-2007-09-0630.
Digital Object Identifier no. 10.1109/TPAMI.2008.127.

semantically similar images, or near-duplicates, instead, and learn from their descriptions to annotate the uncaptioned image, with the hope that these near-duplicates will illustrate the major concepts of this image. An intuitive way to discover near-duplicates is to search. On the other hand, since the surrounding text of Web images is generally noisy, data mining techniques can be adopted to de-noise and figure out salient terms or phrases from the search results to annotate the image. These illustrate the key idea of our approach. It not only investigates *to what extent* data can help us recognize images, as the saying "data is the king," but also demonstrates *how efficient* a practical image annotation system can be.

This approach is supported by newly proposed hash encoding algorithms [13], [47], which convert an image into an N-bit bitstream, and based on which, pairwise similarity can be computed in real time. It partly solves the efficiency problem and shines a light on combining visual features into the commercial image search engines, which, in the current stage, are purely based on textual descriptions.

This paper is organized as follows: Section 2 surveys the previous work on image annotation, which provides readers with a general sense of the related work. Section 3 demonstrates our insights on this problem, based on which we proposed our data-driven and model-free approach. Section 4 presents the approach in detail, with evaluations in Section 5 and a few discussions in Section 6. To evaluate how effective our approach is without a query keyword, we conducted a few experiments, which are discussed in Section 7. This paper concludes in Section 8 with an outlook for possible improvements.

2 Related Work

As early work on image annotation, many researchers resorted to users' relevance feedback (RF) to assign labels to a given image. For example, Liu et al. [29] asked the user to label an image in the RF stage and then propagate these labels to all of the positive images suggested by the retrieval system. Shevade and Sundaram [41] improved it by calculating the propagation likelihood based on WordNet [16] synonym sets as well as on image low-level features, and presenting those images that are the most ambiguous to the user for RF.

As for automatic image annotation, most of the researchers worked in two directions: to learn the joint probabilities between images and words or to learn the conditional probability.

In the former case, generative models were proposed. For example, Barnard et al. [2] and Duygulu et al. [12] represented images in blobs and then adopted a statistical machine translation model to translate the blobs into a set of keywords. Blei and Jordan proposed a *Corr-LDA* model [4]. It assumes that there is a hidden layer of topics, which is a set of latent factors, such that words and image regions are independently generated by the topics. It used 7,000 Corel photos and a vocabulary of 168 words for evaluation. Barnard et al. [1] proposed a hierarchical model in which images and co-occurring text are generated by nodes arranged in a tree structure. An image is thus annotated by the text attached to a path from a leaf to the root which achieves the highest score given the visual and textual features of this image. Their model was trained on 16,000 Corel photos with 155 words and annotated 10,000 test images. Li and Wang [27] proposed a 2D multiresolution hidden Markov model to couple images and concepts. They used 60,000 Corel photos with 600 concepts. In one of their recent works, they improved this model and built an interesting real-time annotation system named *Alipr*, which attracted a great deal of attention from both academic and industry. Lavrenko et al. [24] proposed a continuous relevance model which directly associates continuous features with words and achieved significant improvement in performance. Jeon and Manmatha [22] extended this model and built it with 56,000 Yahoo! news images with noisy annotations and a vocabulary of 4,073 words. This is the largest vocabulary ever proposed and they discussed noisy annotation filtering and speeding-up schemes.

As a different kind of approach, Pan et al. [37] leveraged a graph structure. They constructed a two-layer graph whose nodes are images and their associated captions and proposed a random-walk-with-restart algorithm to estimate the correlations between new images and the existing captions. Then, in another work [36], they extended the model to a three-layer graph with image regions added.

In contrast, a few researchers have proposed discriminative models. Chang et al. [9] learned an ensemble of binary classifiers, each for a specific label. Li et al. [25] proposed a Confidence-based Dynamic Ensemble model, which is a two-stage classifier. Carneiro and Vasconcelos [6] attempted to establish a one-to-one mapping between semantic classes and sets of images with the criterion to minimize the error rate. Mori et al. [34] uniformly divided each image into subimages with key words and then applied vector quantization onto visual features of the subimages to estimate which words will be assigned to the new image.

All of the works mentioned above require a supervised learning stage. Hence, the generalization capability is a crucial assessment to their effectiveness. Among them, the online demo of *Alipr* shows its robustness to outliers, although the model was trained on Corel images.

Due to space limitations, we are not able to list all of the previous works. Interested readers can refer to Smeulders et al.' comprehensive survey in 2000 [43] and a recent survey after 2000 by Datta et al. [11] for a better understanding of the area.

Recently, some researchers began to leverage Web-scale data for image understanding [48], [54], [19], [45]. Wang et al. [48] learned an image thesaurus from Web images and their surrounding text to bridging the semantic gap. Yeh et al. [54] identified locations by searching the Internet. Given a picture of an unknown place, they first obtained a small number of visually relevant Web images using content-based search, then extracted a few keywords from the descriptions of these images. A text-based search was successively performed and the search results were further filtered by visual features iteratively.

The disadvantages of [54] are that, due to the efficiency problem, only a few relevant images were retrieved as seeds, while the semantic gap inevitably biases the final

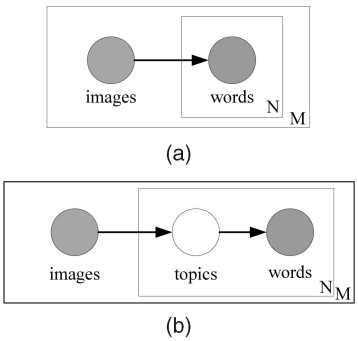

Fig. 1. Generative models for image autoannotation. (a) The two-layer model. Words are directly generated from visual features. (b) The three-layer model. Words are generated from a hidden layer of "topics."

results. However, ignoring all of these vulnerabilities, it is still an important work which pioneers a different kind of solution to image understanding.

Hays and Efros [19] presented another interesting application of leveraging Web-scale data—image completion. Given an image with regions missed, they complemented it by finding similar scenes that contained image fragments which can seamlessly patch up the holes. This approach is entirely data driven. Although the reproduced image may not contain exactly the same content of the original one, it is semantically valid.

Another important work was done by Torralba et al. [45]. They collected 80 million images and attempted to investigate how large of a scale of a data set is required so that the k-nearest neighbor $(\mathrm{k}NN)$ metric is enough to evaluate the visual similarity of two images.

3 Background Theory and Motivation

This section provides our insights into the image auto-annotation problem, which illustrates the motivation as well as the key idea of our model-free image annotation approach.

3.1 Image Autoannotation Models Revisited

Fundamentally, the aim of image auto-annotation is to find a group of keywords $\mathbf{w}^*$ that maximizes the conditional distributions $p(\mathbf{w}|I_q)$, as described in (1a), where I_q is the uncaptioned query image and $\mathbf{w}$ are terms or phrases in the vocabulary. Applying the Bayesian rule, we obtain (1b), where I_i denotes the ith image in the database; hence, $p(I_i|I_q)$ investigates the similarity between I_i and I_q, and $p(\mathbf{w}|I_i)$ evaluates the correlation between I_i and $\mathbf{w}$. This corresponds to the generative model shown in Fig. 1a, in which annotations are generated directly given the images. If we assume that there is a hidden layer of "topics" so that images are represented as a mixture of them and it is from these topics that words are generated, then we obtain a topic model as shown in Fig. 1b, which corresponds to (1c), where t_j represents the jth topic in the topic space:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} p(\mathbf{w}|I_q) \tag{1a}$$

$$= \arg\max_{\mathbf{w}} \sum_i p(\mathbf{w}|I_i)p(I_i|I_q) \tag{1b}$$

$$= \arg\max_{\mathbf{w}} \sum_i \left(\sum_j p(\mathbf{w}|t_j)p(t_j|I_i) \right) p(I_i|I_q). \tag{1c}$$

3.2 A Searcher's Interpretation

Most of the previous generative approaches are covered by these two formulations. Moreover, since the model of (1c) investigates the relationships between images and words in a more exhaustive way, it was generally reported to be more effective than (1b) [1], [4], [33].

In contrast, we interpret (1) from a different angle, specifically, in the language of search and data mining. Intuitively, to a query image, $\mathbf{w}^*$ appears more probably in the contexts of relevant images than irrelevant ones. Hence, we can approach (1b) by generating $\mathbf{w}$ from relevant images instead of the whole data set, while a typical technique to discover relevant images is to search. Let $\Theta_q \doteq \cup_i I_i^q$ denote the set of relevant images $\{I^q_{i=1,\ldots,n}\}$ to I_q. Equation (1b) is reformulated as

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} p(\mathbf{w}|\Theta_q)p(\Theta_q|I_q), \tag{2}$$

where $p(\Theta_q|I_q)$ simulates the search process.

$p(\mathbf{w}|\Theta_q)$, on the other hand, simulates the term generating process which recognizes the semantics in Θ_q. Various techniques can be adopted here. However, unlike the traditional approaches which *learn* it by training either a generative or a discriminative model [1], [4], [33], we interpret it as a data mining process and *mine* $\mathbf{w}^*$ from the textual descriptions of Θ_q. Such a mining process has attracted interest recently in what which are typically called label-based clustering techniques [46], [42], [44], [55].

3.2.1 Brief Introduction to Label-Based Clustering

Differently from traditional document-based clustering methods which measure document distances with the vector space model (e.g., k-means), label-based approaches cluster documents by ranking salient phrases and documents containing a certain salient phrase form a cluster. Obviously, its challenge is in learning the informative and classifiable phrases.

A critical prerequisite of label-based clustering is that all of the documents are relevant in some sense, e.g., they are search results of a query; this ensures reduced noise and diversity. Hence, the most important property of the learned indicative phrases is to detail subsets of the documents. Therefore, it is typically used for facilitating users' browsing [46].

3.2.2 Brief Introduction to SRC [42], [55]

As a typical label-based clustering technique, the *Search Result Clustering* (SRC) approach proposed by Zeng et al. [55] can generate clusters with highly readable names. Concretely, given a set of documents, it learns the clusters in three steps: 1) document parsing and phrase property calculation, 2) salient phrase ranking, and 3) postprocessing.

First, it generates a number of valid phrases (n-grams, $n \leq 3$) from the documents. Stop words are kept so that they can be shown when they are adjacent to meaningful

keywords in cluster names. Five properties are calculated for each phrase, namely, phrase-TFIDF (TFIDF, which is the product of the phrase frequency and the log ratio of inverted document frequency), Phrase Length (which counts the terms in a phrase), Intracluster Similarity (which measures the content compactness of documents indexed by a specific phrase), Cluster Entropy (which identifies the distinctness of a phrase), and Phrase Independence (which measures the independence of a phrase by the entropy of its context).

Second, a linear regression model learned beforehand is utilized to combine these properties into a single salience score and the phrases are ranked in descending order of their scores. The top-ranked phrases are the salient ones. Then, documents containing the same salient phrases are grouped together, which constituted clusters, with the phrases as cluster names. Obviously, a document can be assigned to multiple clusters, a typical characteristic of label-based clustering in which clusters can overlap.

Third, in the postprocessing step, the phrases that contain only stop words or query words are filtered out. Then, clusters are merged to reduce duplicates. The merge strategy is that if two clusters have more than 75 percent documents in common, they will be merged into one cluster; meanwhile, the cluster names are adjusted accordingly and the topmost clusters are output.

The typical advantages of SRC are the following, which make it much more suitable than traditional document-based clustering in tackling our online annotation problem:

1. It emphasizes the efficiency of clustering, which makes it promising in online clustering.
2. It is soft clustering and an image can belong to multiple clusters and can be tagged by multiple salient phrases. This is appealing as images generally have complicated subjects.
3. Each cluster has a name, i.e., salient terms or phrases, which summarizes the common concepts shared by its images. This is a typical advantage, which suggests annotations directly from the textual descriptions.

The SRC technique is currently released online [40].

Based on the analysis above, we propose a novel solution of data-driven image annotation by first retrieving a set of similar images Θ_q with the query (i.e., $p(\Theta_q|I_q)$) and then mining annotations from it (i.e., $p(\mathbf{w}|\Theta_q)$) with SRC, which solves (2). In particular, learning $p(\mathbf{w}|\Theta_q)$ with SRC is to discover topics $\mathbf{t}$ such that

$$p(\mathbf{w}|\Theta_q) = \max_{\mathbf{t}} p(\mathbf{w}|\mathbf{t})p(\mathbf{t}|\Theta_q), \tag{3}$$

where $\mathbf{t}$ is approximated by the "cluster names" output by SRC.

Equation (3) is ensured by SRC's setup of model training. Three human evaluators were asked to label ground truth data for queries which were selected from a Microsoft Live Search query log. These queries are ambiguous queries, entity names, or general terms which tend to contain multiple subtopics. For each query, the evaluators first browsed through all returned pages, then selected from about 200 candidate phrases—which are all meaningful n-grams ($n \leq 3$) in search results—10 "good phrases" and 10 "medium phrases" and scored them with 100 and 50, respectively. The rest of the candidate phrases are "no-interest" ones and were scored zero. The regression model was thus trained based on these scores as well as the aforementioned properties of the n-grams. Obviously, the top-ranked outputs of the model reflect human judgments on good phrases which best represent the topics shared among documents.

In all, we have

$$\begin{aligned}\mathbf{w}^* &= \arg\max_{\mathbf{w}} p(\mathbf{w}|I_q) \\ &= \arg\max_{\mathbf{w}} p(\mathbf{w}|\Theta_q)p(\Theta_q|I_q)\end{aligned} \tag{4a}$$

$$= \arg\max_{\mathbf{w}} \left[\max_{\mathbf{t}} p(\mathbf{w}|\mathbf{t})p(\mathbf{t}|\Theta_q)\right] p(\Theta_q|I_q). \tag{4b}$$

It suggests three critical factors that affect the effectiveness of the approach, namely, the retrieval process $p(\Theta_q|I_q)$, the mining process $p(\mathbf{t}|\Theta_q)$, and the ranking process $p(\mathbf{w}|\mathbf{t})$.

4 The Model-Free Annotation Approach

4.1 Divide-and-Conquer the Semantic Gap Problem

As mentioned before, the semantic gap problem is crucial for image annotation [20]. However, many researchers found that although, individually, the visual features and textual descriptions of images are ambiguous,[1] they tend to not be so when combined together [3], [4], [27], [28].

Thus, we suggest dividing and conquering the annotation problem in two steps: 1) Given a query image, find *one* correct[2] term (or phrase) and 2) given the image and the keyword, find more complementary terms or phrases that interpret the content of the image. The second step is easy to comprehend—with an initial keyword, the ambiguity of the query example is reduced and it is sure to find visually and semantically similar images, at least to some extent.

The requirement in the first step, however, is not as lacking in subtlety as it may first seem. For example, for desktop images, users usually name the folders by locations or event names and, for Web images, generally there are textual descriptions from which the initial keyword can be chosen. On the other hand, the importance of a query keyword's availability decreases as the data set which supports a data-driven approach expands. This intuition is actually proved by Torralba et al. [45]. They found that, when the data set contains 80 million images, simply applying the k-nearest neighbor (kNN) metric to visual features finds satisfying relevant images. Therefore, when the data set is large enough, dividing and conquering the annotation problem may be unnecessary. However, how large is "large enough" is still an open question since it is very difficult, if not impossible, to figure out the data distribution which ensures the effectiveness of kNN in

1. Text ambiguity resulted from synonyms, polysemes, and incorrect parsing. For example, both images labeled as "tiger lily" and "white tiger" are relevant to the query keyword "tiger."

2. Note that we do not require it to be perfect but just "correct," e.g., given an image about the Eiffel Tower, we do not require the word to be "Eiffel" but just "France."

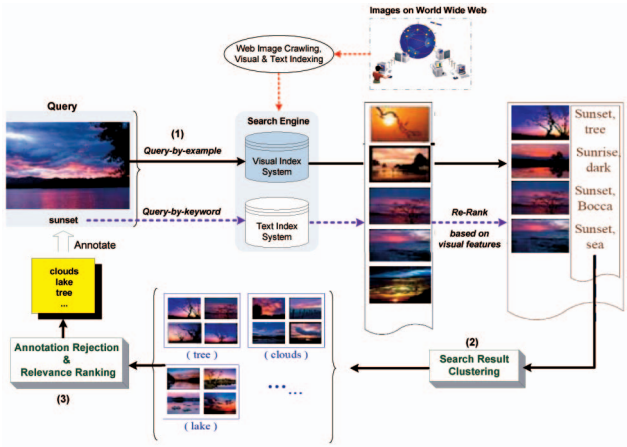

Fig. 2. The flowchart of the model-free process. It contains three steps: 1) search, 2) mining, and 3) ranking. An image and a query keyword ("sunset" here) are inputs. The yellow block highlights the predicted annotations.

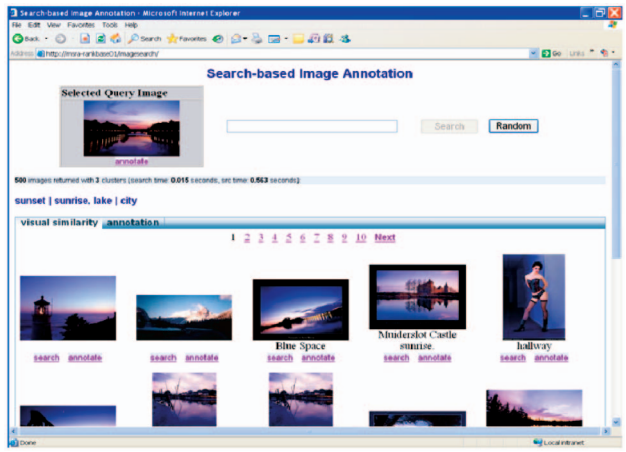

Fig. 3. UI of the annotation system. The gray block highlights the query image. Statistics and predicted annotations are shown below it. The tab form contains image search results which are used for annotation mining.

visual space. Hence, the divide-and-conquer proposal is still competitive, which we believe will help push image understanding a step forward.

We will not discuss the solution to the first step in this paper, leaving it as future work, and tackle the problem of the second step.[3] Thus, the problem is reformulated as

$$\begin{aligned}\mathbf{w}^* &= \arg\max_{\mathbf{w}} p(\mathbf{w}|I_q, w_q) \\ &= \arg\max_{\mathbf{w}} p(\mathbf{w}|\Theta_q) p(\Theta_q|I_q, w_q) \\ &= \arg\max_{\mathbf{w}} \left[\max_{\mathbf{t}} p(\mathbf{w}|\mathbf{t}) p(\mathbf{t}|\Theta_q)\right] p(\Theta_q|I_q, w_q),\end{aligned} \tag{5}$$

where both a query keyword w_q and a query image I_q are provided to trigger the search process.

4.2 Sketch of the Model-Free Annotation Approach

Fig. 2 shows the flowchart of the proposed approach. As a prerequisite of data-driven approaches, 2.4 million photos (mostly natural scenes) were crawled from the Web, the details of which will be given in Section 4.2.1, and then the corresponding visual and textual index files were built to realize the real-time process, for which we give the details in Section 4.2.2.

A natural way to retrieve visually similar images is Query-By-Example (QBE). However, when the image database contains millions or billions of images, image retrieval based on pairwise euclidean distance computation is too time consuming and is thus impractical. We solve this problem in this way: First, Query-By-Keyword (QBK) retrieval is conducted ahead of QBE, which filters out "semantically" dissimilar images in real time. Intuitively, it reduces the image space and, hence, dramatically saves time for the successive QBE retrieval. Second, instead of measuring euclidean distances, we encode image visual features into hash codes which are binary bitwise and efficient distance measures such as Hamming distance can thus be leveraged. We detail this step in Section 4.2.3. These constitute the search stage (labeled by "1" in Fig. 2) and visually and semantically similar images are thus obtained.

The next step is to mine a few terms or phrases from the textual descriptions of these images, which is the mining stage (labeled by "2" in Fig. 2). As described in Section 3.2, we adopt SRC [55] for this task. The cluster names output by SRC are assumed as the topics $\mathbf{t}$ in (5).

Then, in the ranking stage, we propose a simple but effective *relevance ranking and annotation rejection* approach to select and rank the mined cluster names and use the top ones as the output annotations, which simulates $p(\mathbf{w}|\mathbf{t})$. This step is necessary since: 1) SRC [55] is purely based on textual features while it ignores visual features, so the learned cluster names may contain incorrect annotation words (we will investigate this problem in our future work) and 2) the released SRC tool generates about 20 clusters, which is too large a number to promise high precision in our scenario.[4] We detail this step in Section 4.2.3.

This step corresponds to a soft annotation approach since the cluster names generated by SRC vary in length. Thus, even if we fix the number of top clusters, the number of phrases is not fixed.

The user interface is shown in Fig. 3. It supports various query submission schemes—the user can either upload a query image or submit a query term and select an image from its QBK results or click on the "Random" button to randomly choose an image from the database.

The gray block highlights the query image and, when the label "annotate" is clicked, the annotation process starts. Statistics such as time cost and the number of similar images returned are highlighted by the blue bar beneath the query image. As for the real example shown in Fig. 3, 500 images are retrieved from the 2.4 million Web images and the time cost of the search and mining process is 15 and 563 ms, respectively, which is real time. The terms in blue are the predicted annotations, which are cluster names

3. Although we illustrate our approach by assuming that a query keyword is given, in Section 7, we show that, without query keywords, i.e., adopting the traditional image annotation framework in which only query images are given, our method still achieves satisfying performance.

4. Although SRC accepts user-assigned numbers to indicate how many categories are preferred, 20 is a suggested number that statistically produces the highest performance.

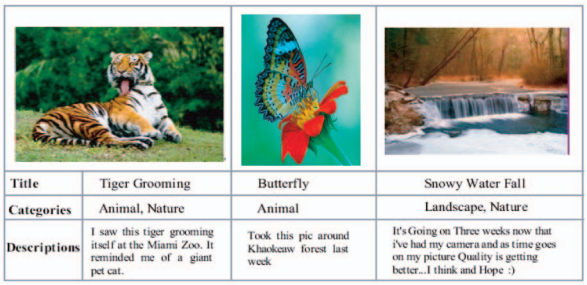

Fig. 4. Three examples of the 2.4 million photos which are generally of high quality and have comments.

yielded by SRC [55] and are separated with the sign "|". In Fig. 3, three terms passed the annotation rejection step. The tab below the form displays a subset of the 500 relevant images whose textual descriptions were part of SRC.

In the following sections, we illustrate the technical details as well as the engineering design which ensures effective annotation in real time.

4.2.1 Crawling a Large, High-Quality Data Set to Ensure an Effective Data-Driven Approach

Since Web images are of various quality and their surrounding text is generally noisy, not just any Web images will help [22] and, hence, a postprocessing step is often required [26], [51]. However, the main purpose of this paper is not data set collection, so we did not put our effort into collecting a high-quality data set as [26] does.

Hence, we simply crawled 2.4 million images from a few photo forums. The advantages are that

1. photos on these forums generally have high resolution since they are taken by photographers,
2. when uploading their work, the photographers will provide comments which more or less describe the semantics of these photos,
3. most of the photos are natural scene images which have simpler semantics compared to artificial ones or portraits like the Corbis data set, and
4. there are lots of images sharing the same semantics but with diverse appearance, which ensures a higher generalization capability of the annotation model than when trained with Corel images.

Three examples from the 2.4 million photos are shown in Fig. 4. We can see that, although the descriptions are noisy, they more or less hit some content in the corresponding images (e.g., "tiger" in the first example) or suggest terms (e.g., "forest") that statistically co-occur with the main objects.

Table 1 provides a few statistics on the 2.4 million photos. The dictionary contains 362,669 words and each photo is commented with 19.6 words on average; to our knowledge, it is to date the largest database used for image annotation, especially for real-time annotation.

4.2.2 The Promise of Real Time

The obstacles to real-time annotation in our case include the content-based retrieval process and handling of the 2.4 million images. We solve the first problem by mapping image visual feature vectors into hash codes and the second one by setting up a distributed system.

TABLE 1
Statistics on the 2.4 Million Data Set

No. of images indexed	2,409,609
No. of images having titles or comments	2,142,156
No. of distinctive words	362,669
No. of distinctive words with frequency ≥ 10	45,438
Average doc length of "title" field	2.3
Average doc length of "comment" field	17.3

Accelerating search through hash encoding. Typically, the similarity of two images is measured in euclidean space. However, since visual features are generally of high dimension, measuring pairwise euclidean distance becomes a bottleneck even if the search space is greatly reduced by text-based search given the query keyword. An effective way to accelerate the content-based search process is to compress the images. Some previous work proposed vector quantization techniques, for example, [56] segments an image into regions and quantizes the regions to obtain a few *keyblocks*. The inverted index technique is thus naturally applied onto such keyblock-based discrete representations and, thereby, the content-based retrieval problem is converted seamlessly into a text-based one which greatly improves the retrieval efficiency.

In this paper, we adopt another technique, named image hash code generation (HCG) [13], [47], since it is more scalable than vector quantization methods [56] and, hence, is a better fit for large-scale databases.

Related work on hash encoding. The HCG algorithms [13], [47] were originally proposed to detect visually identical images. The basic idea is: Suppose that visual features are mapped into bitstreams, with higher bits representing the more important content of an image and lower bits the less important content. Obviously, duplicates should have equal hash codes measured efficiently by the Hamming distance (i.e., the "AND" operation); as for near-duplicates, the "more" highest bits in common, the more probably two images are alike.

The HCG algorithm proposed in [13] is shown in Fig. 5. First, it transforms a color image into gray scale and divides it evenly into 8×8 blocks. Each block is represented by its average intensity so that we obtain an 8×8 matrix with each element I_{ij} defined as

$$I_{ij} = \frac{\sum_{x=0}^{w-1} \sum_{y=0}^{h-1} Int(x,y)}{w \times h}, \tag{6}$$

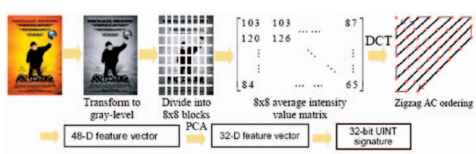

Fig. 5. Hash code generation [13].

Input: feature vector $Y = \{y_i\}$: i ranges from 0 to n-1
For each element y_i
  If $y_i > 0$, the ith bit of *signature* is set to 1
  Else the ith bit of *signature* is set to 0
End
Output: *signature*

Fig. 6. Hash encoding algorithm [13].

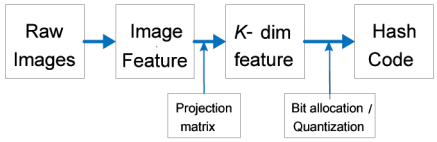

Fig. 7. Hash code generation [47].

where $Int(x, y)$ is the intensity of the (x, y)th pixel and w and h denote the block width and height, respectively.

Two-dimensional DCT transformation [38] is then applied to M. The DC coefficient, which is the average of M, is omitted to eliminate the effect of luminance. The remaining AC coefficients are zigzagged into a sequence and the first 48 coefficients in lower frequencies are collected as the feature vector X_m of the original image, which is further mapped to an n-dimensional vector Y_n by a PCA [23] model P^T trained on 5,500 Web images, shown as

$$Y_n = P^T X_m. \tag{7}$$

The hash code is then generated from Y_n using the encoding algorithm shown in Fig. 6. The intuition behind PCA-based dimension reduction is that the PCA space is essentially a rotated version of the original feature space. If all of the feature dimensions are kept, it draws the same conclusion on data points' distances as measured in the euclidean space. Moreover, since dimension reduction using PCA is achieved by cutting off the low variance dimensions, the information loss is small and can be ignored, while, on the other hand, the hash code-based matching can be significantly speeded up.

Wang et al. [47] proposed a similar idea, but it is more efficient, as shown in Fig. 7. Images are hierarchically divided into blocks and average luminance is used as features. The transformation model is still PCA, but it was trained on 11 million *iFind* [8] images.

Mapping visual features to hash codes. The same technique can be applied to accelerate the search process with low information loss. Recall that the higher bits represent the more important content in an image, so, if two hash codes have equal n highest bits, they can be indexed by the same key. Henceforth, we can significantly speed up the search process in visual space by creating inverted indices based on the hash codes.

In our approach, we use 36-bin color Correlogram [20] instead of intensity or luminance [13], [47] as the original visual features. The reasons are twofold: 1) The aim of [13], [47] is to find duplicate images which are identical both in layout and in content; hence, the luminance of gray-scale images is an effective feature. However, our goal is to find relevant images—images that may not have the same layout and content but share similar concepts. Hence, to keep the color, texture, and shape, etc., properties is important in our case. 2) Correlogram is also a widely used feature in CBIR [40], [7], [10] which simultaneously takes into account color and shape.

Since 3-channel (e.g., RGB) instead of 1-channel [13] features are used in our case, 2D DCT transformation is inapplicable here. Hence, we adopted Wang's approach [7] and mapped the 144-dimensional Correlogram features of all of the 2.4 million photos into 32-bit hash codes, with their higher bits representing more informative content. This is an offline approach.

Hash code-based image retrieval. We designed four distance measures as below, considering the retrieval precision and practicality for an online approach. The results are given in Section 5:

- *Hash code filtering plus euclidean distance* (HD-EuD). If the highest n bits of an image exactly match those of the query image, we keep this image and rank it according to its euclidean distance on Correlograms to the query. Hash encoding here is leveraged as a fuzzy filtering method. The intuition is that hash encoding may introduce error, so we use it as a preprocessing step and rely on the original features to obtain and rank search results. In our experiments, $n = 20$.
- *Hamming distance* (HD). This is the most efficient measure which counts the number of different bits of two hash codes.
- *Weighted Hamming distance* (WHD). HD assumes that all bits have equal weights. However, it is intuitively beneficial to encode the importance of a bit into the distance metric and to give the higher bits larger weights. We propose the weighting function as below: We evenly divide the 32 bits into eight bins[5] and weight the Hamming distance on the ith bin by 2^{8-i}, $1 \le i \le 8$. Obviously, such a weight magnifies the difference on the higher bits, which coincides with the hash encoding scheme. It is simple but effective, as proven in our experiments.
- *Euclidean distance* (Corr-EuD). We also compute euclidean distance on Correlograms as a baseline to evaluate the performance of hash codes.

We rank the images in the ascending order of their distances to the query and return the top$N$ as search results.

Efficiently handling the large-scale database with distributed computing. Hash encoding is not enough for real-time annotation when the database contains millions or, especially, billions of images. In this section, we describe our design of a distributed system, which, as shown as an example in Fig. 3, finishes the search process in 15 ms and the mining process in 563 ms on 2.4 million images.

The system architecture is shown in Fig. 8. The key is that the content-based search engine, text search engine, and SRC clustering engine are provided as Web services using the C# Remoting technique. Each service registers a distinct TCP port and is listening to it; when there is a service request, the services accept input variables, perform their own work, and send back the outputs. Besides, by building up this distributed system, both visual and textual features

5. "8" is an experiential value giving the best performance in our experiments.

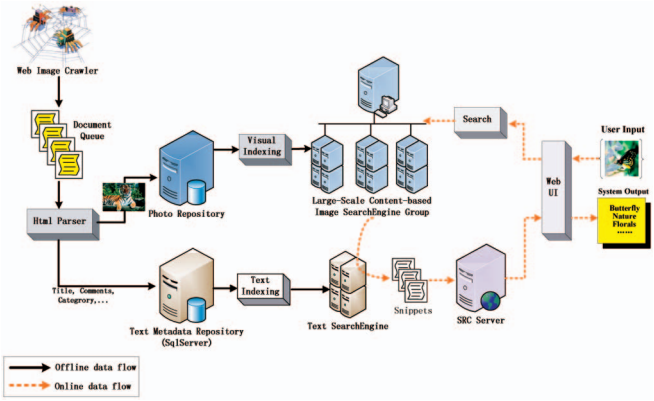

Fig. 8. Distributed system architecture. The search and clustering engines are implemented as Web services.

can be kept in server machines' memories so that no disk I/O is requested, which obviously is quite efficient and is easy to scale up.

4.2.3 The Control of SRC Outputs

As mentioned in Section 4.1, we use the SRC [42], [55] to learn the topics $\mathbf{t}$ in (3) given the visually and semantically similar search results.

A problem is that SRC requires users to determine the number of clusters $|n_{src}|$, which is difficult without observing the data. We use an experiential algorithm as (8) to select $|n_{src}|$:

$$|n_{src}| = \max(|\Theta^*|/200, 4), \tag{8}$$

where $|\Theta^*|$ is the number of search results. An empirical value of cluster size suggested by SRC is 200 to ensure the saliency of learned cluster names. If $|\Theta^*|$ is too small, SRC tends to group all images into one or two clusters and, hence, the clusters may be too diverse to produce meaningful cluster names. To avoid this, we force the algorithm to output at least four clusters, while 4 is another empirically selected parameter.

On the other hand, because SRC extracts all n-grams $(n \leq 3)$ as its candidate key phrases, if $|\Theta^*|$ is too large, the time cost of SRC will become too expensive for an online service. Hence, we require $\max|\Theta^*| = 2{,}000$.

The final step is annotation rejection. As discussed in Section 4.2, it is improper to output all cluster names yielded by SRC as predicted annotations, so annotation rejection is necessary to control the outputs. Two criteria are analyzed in our current implementation:

- *Maximum cluster size criterion* (MaxCS). This criterion uses the number of images in a cluster to score the corresponding cluster name. The highest scored ones are assigned to the query image as the final annotations. This is equivalent to the Maximum-a-Posteriori (MAP) estimation which assumes that "the majority votes for the truth," and, since the cluster names are learned from relevant images, intuitively, the larger cluster covers the more important content of the query image.
- *Maximum average member image score criterion* (MaxIS). This criterion uses the average similarity of the images inside a cluster to score it. The idea is that the smaller the intravariance of a cluster, the more probable it is that it illustrates the content of the query image.

We merge the names of the top ranked clusters by removing the duplicate words and output the results,[6] which closes the entire system process.

TABLE 2
Queries from Google

Apple,Beach,Beijing, Bird, Butterfly, Clouds, Clownfish, Japan, Liberty, Lighthouse, Louvre, Paris, Sunset, Tiger, Tree

TABLE 3
Queries from the University of Washington

Australia, Campus, Cannon beach, Cherries, Football, Geneva, Green lake, Indonesia, Iran, Italy, Japan, San juan, Spring flower, Swiss mountain, Yellowstone

5 Evaluation

To evaluate the effectiveness (i.e., high annotation precision) and efficiency (i.e., low time cost) of our approach, we conducted a series of experiments based on two data sets. One is an open data set, of which 30 images from 15 categories (as shown in Table 2[7]) are randomly collected from the Google image search engine [18]. To give a more objective evaluation of the effectiveness, we deliberately selected a few vague query keywords, e.g., "Paris" for "Sacre Coeur" images. Because no ground truth labeling is available, we manually evaluated the retrieval performance on this data set.

The other testing data set is a benchmark CBIR database provided by the University of Washington (UW).[8] It contains 1,109 images and each has about five tags on average and we use the UW folder names as query keywords (see Table 3). A problem of this data set is that not all contents are annotated. Therefore, we also provide the manually revised results to accept synonyms and correct annotations omitted in UW labels for a fair evaluation.

5.1 Experiments on Google Images

5.1.1 Performance Measure

Evaluation of image autoannotation approaches is still an open problem. Many different approaches have been proposed, e.g., Blei and Jordan [4] use annotation perplexity

6. Note that it is actually soft annotating, i.e., the set w is not fixed in size. The reasons are twofold: First, $|n_{src}|$ depends on the number of search results, as given in (8), and, second, SRC cluster names are not of fixed length. As shown in Fig. 3, there are three cluster names (separated by "|"), with the first and the third containing only one word, respectively, and the second containing two words separated by ",".

7. Since most of the photos in this database are natural images and very few, if any, are artificial images, the testing data set shown in Table 2 mainly contains queries related to "nature." However, it does not mean that the proposed technique is ineffective for artificial images. We show this with the query "Apple." As shown in Fig. 11, the painting of an apple is reasonably annotated with "studio, kitchen, fruit, color."

8. http://www.cs.washington.edu/research/imagedatabase/groundtruth/.

to measure the quality of the models, which is equivalent algebraically to the inverse of the geometric mean per-word likelihood. A few researchers use retrieval precision and recall [4], [6], [21], [29]. In particular, the entire data set is separated into two subsets, one for training and the other for testing, such that images in these two subsets share almost the same vocabularies and data distributions. In the testing stage, a few keywords are selected from the vocabulary to query the annotated testing data set. For a word w, let G be the ground truth relevant images and L be the system outputs; retrieval precision and recall are defined as

$$Precision = \frac{|L \cap G|}{|L|}, Recall = \frac{|L \cap G|}{|G|}, \quad (9)$$

i.e., precision evaluates the proportion of relevant images in the retrieval results, while recall calculates that of the relevant images in all of the relevant images contained in the database.

Barnard et al. [1] directly evaluate the annotation accuracy. Moreover, they propose taking into account incorrect tags and adding an explicit penalty on them. Also, they suggest normalizing the correct and incorrect outputs, which results in a so-called *normalized score* (NS) measure:

$$E_{NS} = r/n - \omega/(N - n), \quad (10)$$

where N is the vocabulary size, n is the total number of tags, and r and ω are the number of correct and incorrect tags, respectively. We have $-1 \leq E_{NS} \leq 1$.

All of the criteria described above require ground truth labels, which is impractical for Web image-based approaches. In the latter case, manual assessment is generally adopted [28], [54].

Since no ground truth is available in our approach, we use the modified NS measure

$$E = (r - \omega)/m, \quad (11)$$

where m denotes the number of predictions. Obviously, we still have $-1 \leq E \leq 1$.

5.1.2 System Effectiveness

Fig. 9 shows the trend of E as the similarity weight changes.

The similarity weight, multiplied by the average visual similarity of image search results, serves as a threshold to filter out irrelevant images in the content-based search stage and the rest will be transferred to SRC mining. Since this parameter determines Θ^* in (8), it directly affects the predicted annotations.

The intuition of this threshold is that, since image similarities vary greatly, a hard threshold cannot promise high performance on any queries. In contrast, our soft threshold is query-dependent and on-demand.

The green square curve in Fig. 9 corresponds to the baseline method, which uses only text-based search to retrieve images so that the search results are not necessarily visually similar. Its performance is, hence, greatly biased by ambiguous and sparse textual descriptions. Meanwhile, since no visual features are available, only MaxCS criterion (see Section 4.2.3) is supported. Figs. 9a and 9b show the trends of E with the MaxCS criterion and the MaxIS criterion, respectively. All four distance measures, i.e., HD, WHD, HD-EuD, and Corr-EuD, were compared.

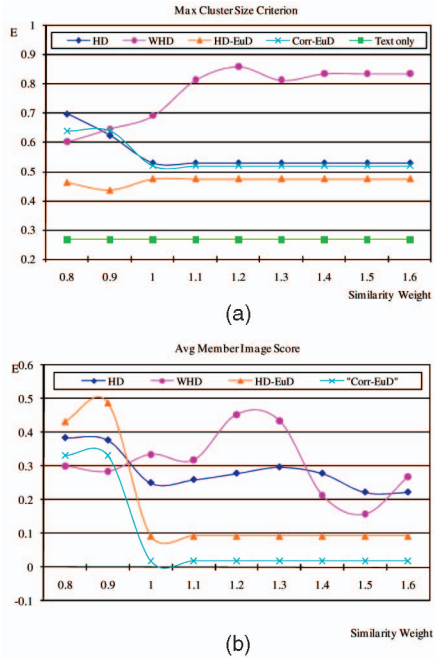

Fig. 9. Annotation precision measured by (11). The WHD measure performs the best. (a) Precision with respect to maximum cluster size criterion. (b) Precision with respect to average member image score criterion.

From Fig. 9a, we can see that WHD (purple dot) performs the best. This is reasonable since it emphasizes the important content in an image while it deemphasizes the unimportant ones.

It is interesting that HD (blue diamond) is comparable to Corr-EuD (cyan cross). This suggests that the information loss of hash encoding has little effect on this data set or, say, it is effective.

Another interesting result is that HD-EuD (orange triangle) performs badly. It may be due to the fact that using the highest 20 bits is too coarse to distinguish relevant images from irrelevant ones.

All of the distance measures are superior to the baseline method, which means that visual features are also important for image understanding.

The MaxIS criterion generally performs worse than the MaxCS criterion, as illustrated in Fig. 9. A possible reason is that SRC clusters images purely based on their surrounding text and, hence, images in one cluster may vary greatly in their visual appearances. Obviously, this does not affect MaxCS criterion but MaxIS, as the latter one uses visual similarity to score the clusters.

5.1.3 A More Comprehensive Performance Measure

We believe that, in the scenario of Web image-based annotation without a vocabulary, to simply differentiate "correct" tags from "incorrect" ones is not enough to evaluate the effectiveness of an annotation system. Specifically, it is better to distinguish "perfect" predictions (those hit specific objects in an image) from "good" ones (correct ones but not perfect, e.g., hypernyms of a "perfect"

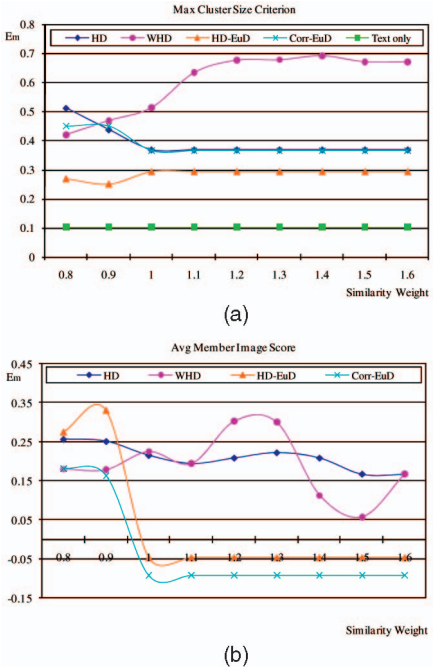

Fig. 10. Annotation precision measured by (12). WHD measure performs the best. (a) Precision with respect to maximum cluster size criterion. (b) Precision with respect to average member image score criterion.

prediction). An example of perfect annotation is "tiger" for the first image in Fig. 4 and "France," in contrast, is a good one for an image of the Eiffel tower. The "bad" annotations are those which are irrelevant to the content of the query image. Taking all of these types of predictions into consideration, we propose (12) as a strict and comprehensive measure:

$$E_m = (p + 0.5 \times r - \omega)/m, \tag{12}$$

where m denotes the number of annotations predicted. p, r, and ω indicate the number of "perfect," "good," and "bad" annotations, respectively. Note that, to emphasize our preference for "perfect" annotations, we punish the "good" ones with a lower weight 0.5. Obviously, when all predictions are "perfect," $E_m = 1$, while, if all are wrong, $E_m = -1$.

5.1.4 System Effectiveness with the New Performance Measure

Intuitively, E_m will be smaller than E with the same experimental settings, as shown in Fig. 10. They resemble similar trends, but the best performances were achieved with different parameters. Besides, from the difference between E_m and E, we can figure out that, for WHD, 30 percent correct predictions are "good" ones.

Fig. 11 provides a few examples of the annotation results. The boldfaced keywords are the queries. Obviously, our approach detected correct annotations and, most of the time, perfect ones.

5.1.5 System Efficiency

We have provided the readers with a first sense of the efficiency in Fig. 3; here, we conducted one more experiment

Fig. 11. A few examples of the predicted annotations.

to evaluate the efficiency statistically. For all of the queries, we collected their search results, which are about 24,000 images on average, and tested the efficiency of the four distance measures: HD, WHD, HD-EuD, and Corr-EuD. The hardware environment is a computer with one Dual Intel Pentium 4 Xeon hyper-threaded CPU and 2 Gbyte memory. The time cost for computing the distance between each of the 24,000 images and the query image, as well as ranking them accordingly, is shown in Fig. 12, which cost is 0.034, 0.072, 0.051, and 0.122 s, respectively. That is, Corr-EuD is nearly four times slower than HD.

HD-EuD is the second efficient. It is because most of the images are filtered out by hash code matching, which has O(1) computation complexity. Time cost for this measure is consumed by the euclidean distance calculation afterward.

5.2 Experiments on UW Data Set

We show the experimental results on the benchmark UW database in this section. All of the 1,109 images are used as queries.

Because ground truth is available, we use precision and recall in (9) as the evaluation criteria. However, differently from the previous image retrieval-based evaluation methods [4], [6], [21], [29], we directly compute the precision and recall on the predictions, i.e., G in (9) represents ground truth tags and L denotes predictions. The effect of the two annotation rejection criteria, MaxCS and MaxIS, as well as the four distance measures on the average performance of all the queries is shown in Fig. 13. Again, WHD performs the best.

An interesting point is that the MaxIS criterion now works better. This is because few images in our 2.4 million photograph database are visually similar to the UW images and the UW images of the same category share similar visual appearance, while MaxIS strategy helps to rank the

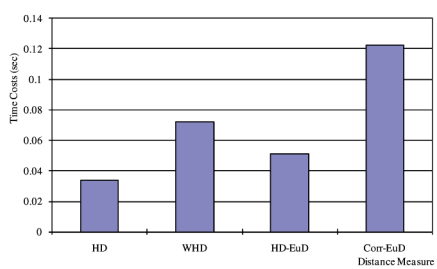

Fig. 12. Search efficiency evaluated on pairwise distances on 24,000 images.

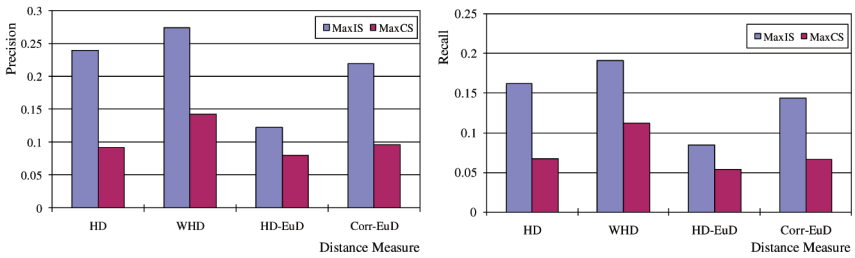

Fig. 13. Precision and recall of the two annotation rejection criteria versus the four distance measures with all images in UW data set as queries.

clusters whose images are visually more relevant to the query higher and thus is less biased by the irrelevant textual descriptions.

It is worth mentioning that the real performance of our system is actually much better than is shown in Fig. 13 since the evaluation shown in Fig. 13 did not regard synonyms, e.g., "beach" and "coast," and semantically relevant keywords, e.g., "Geneva" and "Switzerland," as correct answers. Moreover, the UW ground truth annotations are incomplete, i.e., for most of the images, some content is ignored; however, the above evaluation regards the corresponding predictions as incorrect even if they correctly describe content just because they do not appear in the ground truth. To correct this, we manually examined the results of 100 randomly selected queries. The corrected precision and recall are 38.14 percent and 22.95 percent, respectively, nearly 12 percent precision improvement, with WHD measure, MaxIS criterion, and the similarity weight 1.2.

Fig. 14 shows four examples which hit no UW ground truth tags and, hence, were assumed incorrect but are indeed correct answers.

Note that we adopt no supervised learning stage, while UW images are "outliers" to our database from which few relevant images could be found; the task is thus much tougher for us than for the previous approaches. Moreover, most previous work selects training and testing data from the same data set and the training data set is generally much larger than the testing one, e.g., Barnard et al. [1] and Blei and Jordan [4] use 4,500 Corel images for training and 500 images for testing, and the performance is still around 20 percent to 30 percent. This suggests that our system is more effective and robust.

6 Discussion

Compared to traditional computer vision and machine learning approaches which build generative or discriminative annotation models, we propose a novel model-free approach which investigates how effective a data-driven technique can be and suggest automatically annotating an uncaptioned image by mining its search results. It has at least three advantages:

1. no training data set and supervised learning are required and, hence, the lack-of-training-data problem is avoided;
2. it requires no predefined vocabularies, while vocabulary construction is still an open research topic [35];
3. the Web provides us diverse images sharing the same semantics, which ensures a high generalization capability or the practical usage of the proposed approach; this is noticeably superior to using Corel images, which has very low intraclass variation and incomplete descriptions; and
4. our approach is highly scalable and very robust to outlier queries.

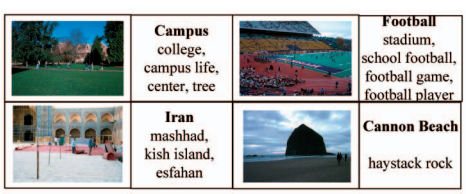

Fig. 14. Four examples from the UW database and their predicted annotations, which hit no UW ground truth tags but are indeed correct answers.

7 Annotating with Visual Queries Only

We would like to investigate in this section how effective the model-free approach is without a query keyword. In this case, image search results are obtained purely based on QBE retrieval. Three series of experiments on Google, UW, and Corel images are conducted.

7.1 Experiments on Google Images

We used the same queries as in Fig. 11 and the results are shown in Fig. 15. We can see that, although the semantic gap problem degrades the performance, satisfying results were still achieved. Moreover, the average time cost is 0.28 s to annotate an image. Note that the absence of textual filtering process results in slower speed and degraded performance.

7.2 Experiments on UW Data Set

To have a comprehensive comparison with the previous approach which is initialized by a query keyword, we evaluate the performance on UW images using the same precision and recall criteria as in Section 5.2. Fig. 16 shows the best performance against the number of images $|\Theta^*|$. AE-Web and AE-UW use all of the 1,109 images as query and adopt automatic evaluations, i.e., a prediction is correct only if it hits a ground truth tag. HC-Web and HC-UW are based on 50 randomly selected UW queries and manual assessment was applied. Moreover, AE-Web and HC-Web adopt unlimited vocabulary suggested by the 2.4 million

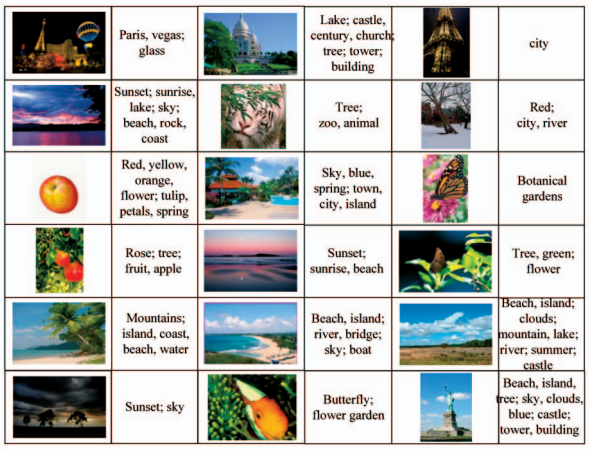

Fig. 15. A few examples of the annotation results yielded by the model-free and data-driven image annotation approach.

Web images, while AE-UW and HC-UW restrict SRC to select words from the UW vocabulary.

A sharp precision degradation was obtained for the AE-Web method. The reasons are that, in addition to the outlier problem and strict evaluation, it was also greatly affected by the semantic gap problem since no query keyword was provided to filter out those semantically irrelevant images.

To reduce the cost of manual labeling, 50 images were randomly selected. Human check shows that the real precision and recall of the annotation system (i.e., HC-Web) are 0.18 and 0.22, which are 157.1 percent and 100 percent improvements, respectively. Moreover, by limiting the vocabulary (i.e., HC-UW), both precision and recall were significantly improved ($p = 0.26$, $r = 0.18$). This implies that higher performance is more easily achieved in a closed data set with a fixed vocabulary. Among all of these results, HC-Web has the highest recall, which benefited from the large-scale data, and, meanwhile, precision drops a little due to the increased probability of incorrect predictions.

The 50 images are associated with 73 unique words in the UW vocabulary, while our approach detected nine correct new words for them, such as city, summer, and rose. It further shows the strength of the annotation system, i.e., higher recall and larger vocabulary. Fig. 17 shows four illustrative examples.

7.3 Experiments on Corel Images

Since it coincides with the traditional annotation strategy that only one query image is given, we are able to compare

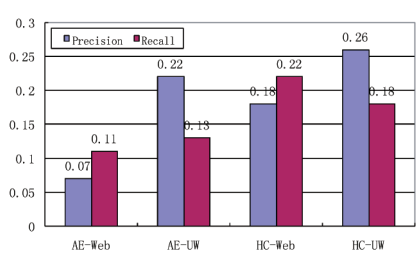

Fig. 16. Performance of UW images without a query keyword. "HC-" methods use 50 randomly selected images.

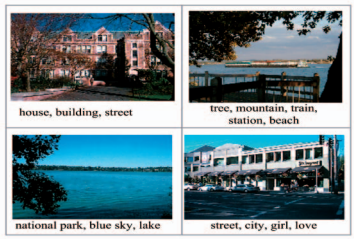

Fig. 17. Four UW examples and our predictions, most of which are not in the UW vocabulary.

this approach (SBIA) with previous work. The baseline methods are the machine translation model (MT) [12], the multi-instance learning model (MIL) [53], and the cross-media relevance model (CMRM) [21].

To get a fair and objective evaluation, we base our experiment on 5,000 Corel images[9] with 371 keywords and 500 blobs overall. Instead of using the 2.4 million Web images as our database, we indexed 4,500 of the Corel images and set aside the other 500 for testing. The same settings were applied to train and test the MT, MIL, and CMRM models.

The same features as in [12], [53], [21] were used so that the four algorithms had the same input. Search results were obtained through blob-based retrieval and the ranking function was BM25 [39].

The average per-word precision and recall of the four methods are illustrated in Fig. 18 for the best 49 keywords [53], [21]. Clearly, our approach outperforms MT and MIL and is comparable to CMRM; however, note that we have no supervised learning while the other three models do.

7.4 Discussions on Closed Data Set and Open Data Set

We can see that the performance of our approach on Corel data set ($p = 0.39$, $r = 0.49$) is much better than that on the UW data set ($p = 0.18$, $r = 0.22$). A key reason is that the experimental setup is totally different, which suggests that annotating with a fixed vocabulary generally achieves higher precision than with an open one. Specifically, the Corel-based experiment was conducted on a closed data set which labeled the 10 percent testing images with 90 percent close-set images and the strong correlation between these two sets led to good performances. However, intuitively, this is impractical.

In contrast, for the UW data set, we treated it as a black box in the system and annotated the images with the 2.4 million photo forum data, which obviously are weakly correlated to this data set. This is a much closer setting to a real scenario, and annotating an open data set without any prior knowledge is obviously a very challenging problem.

8 Conclusion

Compared to the previous annotation approaches, which built up generative or discriminative annotation models, we proposed a novel attempt of model-free image annotation.

9. http://www.cs.arizona.edu/people/kobus/research/data/eccv_2002/.

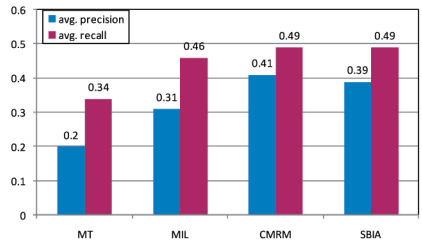

Fig. 18. Performance comparison with MT [12], MIL [53], and CMRM [21]. SBIA is our method.

It not only evaluates how much *data* can help us in image understanding, but also illustrates that *search*, as an important technique itself, can help us in various applications. We collected 2.4 million photo forum images with their textual descriptions for this task.

The entire process contains three steps: 1) the search stage, which retrieves visually and semantically similar images given the uncaptioned images as queries, 2) the mining stage, which applies a label-based clustering technique called SRC to mine key phrases from the search results as candidate annotations, and 3) the annotation rejection stage, which postprocesses the candidate annotations to ensure the preciseness of the final outputs.

Compared with the previous work in this research area, our method saves labor in both training data collection and vocabulary construction and is highly scalable and robust to outliers. Moreover, to our knowledge, it is the real-time approach that handles the largest data set to date.

To ensure real-time annotation, two key techniques are leveraged—one is to map the high-dimensional image visual features into hash codes; the other is to implement it as a distributed system, of which the search and mining processes are provided as Web services. As a typical result, the entire process finishes in less than 1 s.

Experiments conducted on both an open data set (Google images) and a benchmark data set (UW CBIR database) show that our approach is effective in annotation precision and is practical as not only being vocabulary-free but also tagging images in real time.

There is much room to improve the approach, e.g., as mentioned in Section 4.2, it is necessary and important to propose a data mining technique which simultaneously takes visual features into consideration. Also, we would like to investigate how much performance gain we can obtain by embedding learning approaches into our current implementation.

Acknowledgments

The authors would like to thank Feng Jing, Hua-Jun Zeng, Kefeng Deng, Bin Wang, Le Chen, Richard Cai, and Jiang-Ming Yang for their sincere helps on technical discussions, assistance, as well as system implementations. Many thanks should also be given to the volunteers who helped in manually evaluating the experimental results. This work was done while Xirong Li was an intern at Microsoft Research Asia.

References

[1] K. Barnard et al., "Matching Words and Pictures," *J. Machine Learning Research,* no. 3, pp. 1107-1135, 2003.
[2] K. Barnard et al., "Recognition as Translating Images into Text," *Internet Imaging IX, Electronic Imaging,* 2003.
[3] K. Barnard et al., "Clustering Art," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 434-439, 2001.
[4] D. Blei and M.I. Jordan, "Modeling Annotated Data," *Proc. 26th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 127-134, 2003.
[5] D. Cai et al., "Hierarchical Clustering of WWW Image Search Results Using Visual, Textual and Link Information," *Proc. ACM Int'l Conf. Multimedia,* pp. 952-959, 2004.
[6] G. Carneiro and N. Vasconcelos, "A Database Centric View of Semantic Image Annotation and Retrieval," *Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 559-566, 2005.
[7] C. Carson et al., "Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, no. 8, pp. 1026-1038, Aug. 2002.
[8] Z. Chen et al., "iFind: a Web Image Search Engine," *Proc. 24th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* p. 450, 2001.
[9] E. Chang et al., "CBSA: Content-Based Soft Annotation for Multimodal Image Retrieval Using Bayes Point Machines," *IEEE Trans. Circuits and Systems for Video Technology,* vol. 13, no. 1, pp. 26-38, 2003.
[10] I.J. Cox et al., "The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical," *IEEE Trans. Image Processing,* vol. 9, no. 1, pp. 20-37, 2000.
[11] R. Datta et al., "Content-Based Image Retrieval—Approaches and Trends of the New Age," *Proc. ACM Multimedia Workshop Multimedia Information Retrieval,* pp. 253-262, 2005.
[12] P. Duygulu et al., "Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary," *Proc. European Conf. Computer Vision,* pp. 97-112, 2002.
[13] X. Fan et al., "Photo-to-Search: Using Multimodal Queries to Search the Web from Mobile Devices," *Proc. ACM Multimedia Workshop Multimedia Information Retrieval,* pp. 143-150, 2005.
[14] J.P. Fan et al., "Multi-Level Annotation of Natural Scenes Using Dominant Image Components and Semantic Concepts," *Proc. ACM Int'l Conf. Multimedia,* pp. 540-547, 2004.
[15] J.P. Fan et al., "Automatic Image Annotation by Using Concept-Sensitive Salient Objects for Image Content Represent," *Proc. 27th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 361-368, 2004.
[16] *WordNet: An Electronic Lexical Database,* C. Fellbaum, ed. MIT Press, 1998.
[17] A. Ghoshal et al., "Hidden Markov Models for Automatic Annotation and Content-Based Retrieval of Images and Video," *Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 544-551, 2005.
[18] Google Image, images.google.com, 2008.
[19] J. Hays and A.A. Efros, "Scene Completion Using Millions of Photographs," *Proc. ACM SIGGRAPH,* 2007.
[20] J. Huang et al., "Image Indexing Using Color Correlograms," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* p. 762, 1997.
[21] J. Jeon et al., "Automatic Image Annotation and Retrieval Using Cross-Media Relevance Models," *Proc. 26th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 119-126, 2003.
[22] J. Jeon and R. Manmatha, "Automatic Image Annotation of News Images with Large Vocabularies and Low Quality Training Data," *Proc. ACM Int'l Conf. Multimedia,* 2004.
[23] I.T. Joliffe, *Principal Component Analysis.* Springer-Verlag, 1986.
[24] V. Lavrenko et al., "A Model for Learning the Semantics of Pictures," *Proc. 16th Conf. Neural Information Processing Systems,* pp. 553-560, 2003.
[25] B.T. Li, K. Goh, and E. Chang, "Confidence-Based Dynamic Ensemble for Image Annotation and Semantics Discovery," *Proc. ACM Int'l Conf. Multimedia,* pp. 195-206, 2003.
[26] J. Li et al., "OPTIMOL: Automatic Object Picture CollecTion via Incremental MOdel Learning," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.

[27] J. Li and J. Wang, "Automatic Linguistic Indexing of Pictures by a Statistical Modeling Approach," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 25, no. 9, pp. 1075-1088, Sept. 2003.
[28] J. Li and J.Z. Wang, "Real-Time Computerized Annotation of Pictures," *Proc. ACM Int'l Conf. Multimedia,* pp. 911-920, 2006.
[29] W.Y. Liu et al., "Semi-Automatic Image Annotation," *Proc. Human-Computer Interaction,* pp. 326-333, 2001.
[30] X. Li et al., "Image Annotation by Large-Scale Content-Based Image Retrieval," *Proc. ACM Int'l Conf. Multimedia,* pp. 607-610, 2006.
[31] X. Li et al., "SBIA: Search-Based Image Annotation by Leveraging Web-Scale Images," *Proc. ACM Int'l Conf. Multimedia,* pp. 467-468, 2007.
[32] F. Monay and P. Gatica-Perez, "On Image Auto Annotation with Latent Space Models," *Proc. ACM Int'l Conf. Multimedia,* pp. 275-278, 2003.
[33] F. Monay and P. Gatica-Perez, "PLSA-Based Image Auto-Annotation: Constraining the Latent Space," *Proc. ACM Int'l Conf. Multimedia,* pp. 348-351, 2004.
[34] Y. Mori et al., "Image-to-Word Transformation Based on Dividing and Vector Quantizing Images with Words," *Proc. First Int'l Workshop Multimedia Intelligent Storage and Retrieval Management,* 1999.
[35] M. Naphade et al., "Large-Scale Concept Ontology for Multimedia," *IEEE Multimedia,* vol. 13, no. 3, pp. 86-91, July-Sept. 2006.
[36] J.-Y. Pan et al., "GCap: Graph-Based Automatic Image Captioning," *Proc. Conf. Computer Vision and Pattern Recognition Workshop,* vol. 9, p. 146, 2004.
[37] J.-Y. Pan et al., "Automatic Multimedia Cross-Modal Correlation Discovery," *Proc. 10th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining,* pp. 653-658, 2004.
[38] K.R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications.* Academic, 1990.
[39] S.E. Robertson et al., "Okapi at TREC-3," *Proc. Third Text Retrieval Conf.,* pp. 109-126, 1995.
[40] Y. Rui et al., "Image Retrieval: Current Techniques, Promising Directions, and Open Issues," *J. Visual Comm. and Image Representation,* vol. 10, no. 1, pp. 39-62, 1999.
[41] B. Shevade and H. Sundaram, "Incentive Based Image Annotation," Technical Report AME-TR-2004-02, Arizona State Univ. 2004.
[42] "Search Result Clustering Toolbar in Microsoft Research Asia," SRC, http://rwsm.directtaps.net/, 2006.
[43] A.W.M. Smeulders et al., "Content-Based Image Retrieval at the End of the Early Years," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 22, no. 12, pp. 1349-1380, Dec. 2000.
[44] H. Toda and R. Kataoka, "A Search Result Clustering Method Using Informatively Named Entities," *Proc. Seventh ACM Int'l Workshop Web Information and Data Management,* pp. 81-86, 2005.
[45] A. Torralba et al., "Tiny Images," Technical Report MIT-CSAIL-TR-2007-024, Massachusetts Inst. of Technology, 2007.
[46] Vivisimo(2008), http://www.vivisimo.com, 2008.
[47] B. Wang et al., "Large-Scale Duplicate Detection for Web Image Search," *Proc. IEEE Int'l Conf. Multimedia and Expo,* pp. 353-356, 2006.
[48] X.-J. Wang et al., "Data-Driven Approach for Bridging the Cognitive Gap in Image Retrieval," *Proc. IEEE Int'l Conf. Multimedia and Expo,* no. 3, pp. 2231-2234, 2004.
[49] X.-J. Wang et al., "Multi-Model Similarity Propagation and Its Application for Web Image Retrieval," *Proc. ACM Int'l Conf. Multimedia,* pp. 944-951, 2004.
[50] X.-J. Wang et al., "AnnoSearch: Image Auto-Annotation by Search," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* pp. 1483-1490, 2006.
[51] J. Weijer et al., "Learning Color Names from Real-World Images," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* 2007.
[52] Yahoo! News Search, http://news.search.yahoo.com/news, 2008.
[53] C. Yang et al., "Region Based Image Annotation through Multiple-Instance Learning," *Proc. ACM Int'l Conf. Multimedia,* pp. 435-438, 2004.
[54] T. Yeh et al., "Searching the Web with Mobile Images for Location Recognition," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition,* no. 2, pp. 76-81, 2004.
[55] H.J. Zeng et al., "Learning to Cluster Web Search Results," *Proc. 27th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 210-217, 2004.
[56] L. Zhu et al., "Keyblock: An Approach for Content-Based Image Retrieval," *Proc. ACM Int'l Conf. Multimedia),* pp. 157-166, 2000.

Xin-Jing Wang received the PhD degree from Tsinghua University in 2005. She is currently an associate researcher at Microsoft Research Asia. Her primary research interests include image retrieval, image understanding, pattern recognition, and machine learning.

Lei Zhang received the PhD degree from Tsinghua University in 2001. He is a researcher and a project lead at Microsoft Research Asia. His current research interests include search relevance ranking, Web-scale image retrieval, social search, and photo management and sharing. He is a member of the IEEE and the ACM.

Xirong Li received the BS and MS degrees in computer science from Tsinghua University in 2005 and 2007, respectively. He is currently a PhD student in the Intelligent Systems Lab Amsterdam (ISLA), University of Amsterdam (UvA). He was an intern at Microsoft Research Asia from October 2005 to June 2007.

Wei-Ying Ma received the BS degree from National Tsinghua University, Hsinchu, Taiwan, in 1990 and the MS and PhD degrees from the University of California at Santa Barbara in 1994 and 1997, respectively. He is a senior researcher and a research manager at Microsoft Research Asia, where he has been leading a research group to conduct research in the areas of information retrieval, Web search, data mining, mobile browsing, and multimedia management. He has published five book chapters and more than 100 international journal and conference papers. He is a senior member of the IEEE.