Constructing Concept Lexica With Small Semantic Gaps

Yijuan Lu, *Member, IEEE*, Lei Zhang, *Member, IEEE*, Jiemin Liu, and Qi Tian, *Senior Member, IEEE*

***Abstract*—In recent years, constructing mathematical models for visual concepts by using content features, i.e., color, texture, shape, or local features, has led to the fast development of concept-based multimedia retrieval. In concept-based multimedia retrieval, defining a good lexicon of high-level concepts is the first and important step. However, which concepts should be used for data collection and model construction is still an open question. People agree that concepts that can be easily described by low-level visual features can construct a good lexicon. These concepts are called concepts with small semantic gaps. Unfortunately, there is very little research found on semantic gap analysis and on automatically choosing multimedia concepts with small semantic gaps, even though differences of semantic gaps among concepts are well worth investigating.**

In this paper, we propose a method to quantitatively analyze semantic gaps and develop a novel framework to identify high-level concepts with small semantic gaps from a large-scale web image dataset. Images with small semantic gaps are selected and clustered first by defining a confidence score and a content-context similarity matrix in visual space and textual space. Then, from the surrounding descriptions (titles, categories, and comments) of these images, concepts with small semantic gaps are automatically mined. In addition, considering that semantic gap analysis depends on both features and content-contextual consistency, we construct a lexicon family of high-level concepts with small semantic gaps (LCSS) based on different low-level features and different consistency measurements. This set of lexica is both independent to each other and mutually complimentary. LCSS is very helpful for data collection, feature selection, annotation, and modeling for large-scale image retrieval. It also shows a promising application potential for image annotation refinement and rejection. The experimental results demonstrate the validity of the developed concept lexica.

***Index Terms*—Image retrieval, large-scale, lexica, semantic gap.**

I. Introduction

RECENT years have witnessed a fast development of multimedia information retrieval. Despite continuous efforts in exploring new multimedia information retrieval techniques, the semantic gap—the gap between the expressing power of low-level features and high-level semantic concepts—is still a fundamental barrier. In order to reduce the semantic gap, a promising paradigm of concept-based multimedia search has been introduced into many practical search systems in the past few years. This paradigm focuses on modeling high-level semantic concepts, either by object recognition or image annotation. Among various approaches, the first step is to select a lexicon that is relatively easy for computers to understand and then to collect training data to learn the concepts.

However, the problem of lexicon selection is usually either simplified by manual selection or completely ignored in most previous works. For example, researchers working on object classification and recognition manually defined a number of datasets, including UIUC [1], Caltech 101 [2], Caltech 256 [3], and PASCAL [4]. When choosing concepts to construct these datasets, they implicitly favored relatively "easy" concepts, although it is still very challenging to model them. Other researchers working on image annotation either simply use all the keywords associated with training images (ALIPR [5], SML [6]), or do not impose any limitations to the annotation vocabulary (ESP [7], LabelMe [8], and AnnoSearch [9]). These approaches actually ignore the differences among keywords in terms of semantic gap.

There is no doubt that these efforts make their unique contributions to the standardization of concept corpus, thus allowing the multimedia community focuses ongoing research on a well-defined set of semantics. However, we argue that semantic gaps are actually not uniform in a low-level feature space, and that it is inappropriate to ignore the semantic gap differences. For example, it is well acknowledged that modeling "Europe" is more challenging than modeling "sunset" due to the lack of an effective visual feature that can represent the concept of "Europe". Also, researchers usually choose color features to model concepts like "sunset", but choose local features to model concepts like "building".

Concepts with smaller semantic gaps are likely to be better modeled and retrieved than concepts with larger ones. However, very little research can be found in current literature on quantitative analysis of semantic gap. The open problems remain: *what are the well-defined semantic concepts for learning*, and *how can they be automatically found*? This highlights a critical requirement for establishing an efficient way to "measure" the semantic gap, thus helping find the high-level concepts with small semantic gaps, which should be given high priority for data collection, modeling, and training.

Motivated by this, we focus on two key problems: what are the high-level concepts with small semantic gaps and how can

Manuscript received July 27, 2009; revised November 20, 2009 and January 20, 2010; accepted January 24, 2010. First published March 22, 2010; current version published May 14, 2010. This work was supported in part by the Research Enhancement Program (REP) and start-up funding from the Texas State University, the Army Research Office (ARO) grant under W911NF-05–1-0404, and the Department of Homeland Security (DHS). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ajay Divakaran.

Y. Lu is with the Department of Computer Science, Texas State University, San Marcos, TX 78666 USA (e-mail: yl12@txstate.edu).

L. Zhang is with Microsoft Research Asia, Beijing 100190, China (e-mail: leizhang@microsoft.com).

J. Liu is with Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: liujiemin8715@gmail.com).

Q. Tian is with the Department of Computer Science, University of Texas at San Antonio, San Antonio, TX 78249 USA (e-mail: qitian@cs.utsa.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2010.2046292

they be identified? In other words, which semantic concepts should we focus on first to ensure that they can be well modeled and easily annotated? We answer these questions by proposing a novel framework for automatically constructing concept lexica from a large web-scale image dataset of over 2.4 million images collected from several online photo forum websites [10].

Web images are usually associated with rich textual features, such as filename, title, alt text, and surrounding text [11]. These textual features are much closer to the semantics of the images than visual features. The input titles and comments assigned by users when they upload their photos online are especially good semantic descriptions for images. Therefore, we particularly focus on the collected web images and fully utilize this useful information to measure semantic gaps of images. Then, images with small semantic gaps are identified, and the key concepts learned from their surrounding text are output as the produced concepts.

Since semantic gap analysis depends on both features and content-contextual consistency, in this paper, we also analyze semantic gaps of concepts with different low-level features and different consistency measurements. A set of mutually independent and compensatory lexica are produced. By comparing different lexica with small semantic gaps, we can select more appropriate low-level features as representative features and choose better search schemes for each concept.

To our knowledge, this is the first attempt that quantitatively and automatically identifies high-level concepts with small semantic gaps from a huge repository of well-annotated photographs on the Web. Compared with the limited number of manually selected concepts, the proposed approach is potentially capable of constructing well-defined lexica customized for given feature spaces, and can be very helpful for data collection and concept modeling.

In our design, we request two properties for each desired concept lexicon in the lexica family. First, the words (concepts) in the lexicon should have high occurrence frequency within the descriptions of real-world images, which makes them commonly used concepts. Secondly, the chosen concepts are expected to be visually and semantically consistent. That is, the images associated with these concepts have smaller semantic gaps, making them moderately easy to be modeled for retrieval and annotation.

The contributions of this work can be highlighted as follows.

1) We quantitatively study and formulate the semantic gap problem and propose a novel framework including a measurement of semantic gap and algorithms for dominant concept identification to automatically select visually and semantically consistent concepts.
2) The constructed lexica family shows promising application potential for concept detection, automatic annotation, and multimedia information retrieval. As the chosen concepts are ranked based on their semantic gaps, researchers can either focus on modeling concepts which are visually and semantically more consistent, or concentrate on designing rejection strategies to decline those tough concepts with low confidence.
3) This work also studies lexica construction in different feature spaces with different content-contextual consistency measurements. The developed lexica family is helpful for choosing a feature space and selecting appropriate search methodologies for a given concept. This will explicitly guide the research in concept modeling and provide a possibility for multimodality modeling.

The rest of the paper is organized as follows: Section II introduces the related work. Section III presents the lexicon construction procedure. Section IV gives comprehensive experimental results. Section V describes a family of lexica development. Finally, conclusions and future work are given in Section VI.

II. Related Works

We reviewed work closely related to our motivation for constructing concept lexica with small semantic gaps in the following two perspectives: previous work on lexicon construction and related algorithms applied in our work.

A. Previous Work on Lexicon Construction

Lexicon selection and data collection are essential elements of concept-based image retrieval. Publicly available image databases, such as UIUC [1], Caltech-101 [2], Caltech-256 [3], and PASCAL [4], contain many manually selected concepts for category-level recognition. Web-based annotation tools (ESP [7] and LabelMe [8]) have recently provided a new way of building large annotated database by relying on the collaborative effort of a large number of users [12]. When playing games, players enter labels describing the content of images. A lexicon can be collected from these labels.

In 2006, Large-Scale Concept Ontology for Multimedia (LSCOM) [13] was proposed. LSCOM is an ontology of about 1000 concepts produced based on manually annotating a large corpus of 80 hours of broadcast news video. It was designed to optimize utility to facilitate end-user access, cover a large semantic space, make automated extraction feasible, and increase observability in diverse broadcast news video data sets. LSCOM-Lite is a subset of the full LSCOM, and it contains 39 concepts with annotations over the entire development set of TRECVID [6] 2005 videos. It was selected based on semi-automatic mapping of noun search terms from BBC query logs to WordNet senses. MediaMill challenge concept data [14] is a lexicon of 101 concepts selected by taking LSCOM as a leading example and analyzing extended manual annotations.

However, one important question was not proposed until 2007 by Hauptmann *et al.* [15]. That is, what kinds of concepts are most useful? They took the first step towards answering this question by analyzing TRECVID'05 [6] video archive annotated with the 320 LSCOM concepts. They computed a concept utility measure to gauge how each concept contributes to retrieval. In their work, useful concepts were selected from statistical aspects of concept frequency.

Although the above research proposes some useful concepts, it does not consider the semantic gap information of concepts, and does not propose any automatic way to select concepts with small semantic gaps. The difference of semantic gaps among concepts deeply affects performance of corresponding concept detection. Concepts with small semantic gaps will be better modeled and retrieved. Hence, constructing lexica of

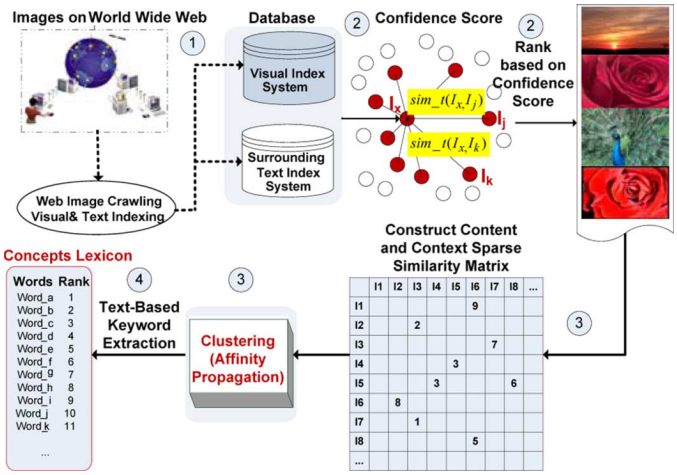


Fig. 1. Framework of LCSS construction: (1) data collection; (2) identifying images with small semantic gaps—calculate semantic gaps of collected images using proposed confidence score, then rank images based on their confidence score and select the top ones; (3) image clustering—construct similarity matrix of selected images and cluster them by affinity propagation; (4) text-based keyword extraction—automatically extract concepts with small semantic gaps from surrounding text of the clustered images.

concepts with small semantic gaps is important and necessary for multimedia information retrieval.

B. Related Algorithms Applied in Our Work

The biggest barrier to selecting concepts with small semantic gaps is that the semantics of images are very difficult to describe. Very little research can be found in the current literature on how to analyze the semantic gap. A search-based image annotation method [9] was recently proposed to use surrounding text of web images to annotate images. Given an image, the search-based image annotation method retrieves a set of similar web images based on visual features. Then it extracts shared keywords from text surrounding these images as the new image's annotation. Motivated by this, we found that semantic information is available for web images from their rich context features, such as title, category, and photographers' comments. This input information, provided when users upload images online, actually describes the images' semantics. Hence, we propose to utilize the context information of web images as their semantic description and design a framework to automatically identify concepts with small semantic gaps. This framework (shown in Fig. 1) contains four steps: 1) data collection—collect a huge number of images from internet that have rich surrounding text information; 2) identifying images with small semantic gaps—define a confidence score to quantitatively measure the semantic gap of these images by using their surrounding text, then rank all images based on the calculated confidence score and select the top images; 3) image clustering—using clustering algorithms to cluster content and contextual similar images together. In this paper, affinity propagation [17] is used for clustering because it can automatically group similar images together very fast, even for large scale data sets, and it does not need to specify the number of clusters. Constructing a content-context similarity matrix is the first step to applying affinity propagation in our work; 4) text-based keyword extraction—design some

Image	Title	Descriptions
	Sea sunset	Sunset at the sea
	Red Rose	A rose in my garden taken June 8th 2002 (My other hobby is rose gardening)…
	The Falls	This is a waterfall that is about 3 miles from my house. It's called The Falls…

Fig. 2. Example images collected from online photo forums. Their title and surrounding descriptions are also listed.

text-based keyword extraction methods to automatically mine concepts from the surrounding text of clustered images. These extracted concepts will construct a concept lexicon with small semantic gaps.

III. Lexicon Construction

A. Data Collection

About 2.4 million web images were collected from five online photo forum sites including Photosig.com, Photo.net, etc. We chose these forum sites because their photos have very high quality and rich textual information such as title and photographer's comments. As shown in Fig. 2, these descriptions cover the content of the corresponding photos to a certain degree.

Semantic gap really depends on low-level features. In this paper, we first extract a 64-dimensional global visual feature vector [16] for all 2.4 million images, which contains three different kinds of color features: six-dimensional color moments in LUV color space, 44-dimensional banded auto-color correlogram in HSV color space, and 14-dimensional color texture moments. More features are tested in Section V. All the collected surrounding texts are stemmed and stop words are filtered out. We build indexes for all images based on low-level visual features and surrounding textual features, respectively.

B. Visual-Textual Confidence Map

All images in a large-scale image dataset with rich surrounding textual information can be located in two different high-dimensional spaces: visual space and textual space.

By definition, semantic gap is the difference between two descriptions by low-level visual features and high-level concepts. If the nearest neighbors of an image in visual space are the same as its nearest neighbors in textual space [as shown in Fig. 3(a)], this image has a small semantic gap because of both consistency. If the nearest neighbors of an image in visual space are dispersed around it in textual space [as shown in Fig. 3(b)], this image has a loose-textual semantic gap. Similarly, if the nearest neighbors of an image in textual space are dispersed around it in visual space [as shown in Fig. 3(c)], this image has a loose-visual semantic gap.

1) Nearest Neighbor Confidence Score: In order to measure the consistency of an image in visual space and textual space, we define a novel nearest neighbor confidence score (NNCS) to

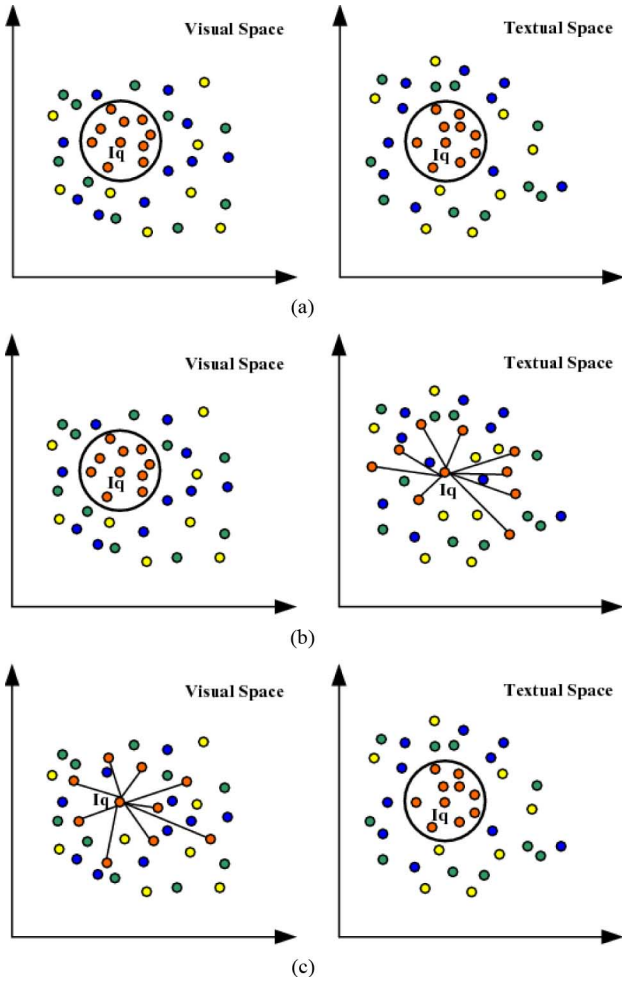


Fig. 3. Consistency in visual space and textual space. (a) Image I_q has consistency in both visual and textual space. Hence, it has a small semantic gap. (b) The nearest neighbors of image I_q in visual space are disperse in textual space. It has a loose-textual semantic gap. (c) The nearest neighbors of image I_q in textual space are disperse in visual space. It has a loose-visual semantic gap.

evaluate the semantic gap between visual (content) and textual (context) features.

Viewing each image as a K-NN classifier, for a particular image I_x, we obtain its K nearest neighbors $\{I_i | i = 1, 2 \ldots, K\}$ based on its visual feature. Assuming I_x and one of its neighbors I_i both have surrounding texts, we can measure their textual description's cosine similarity $sim\_text(I_x,\ I_i)$ based on their textual features. Then the NNCS of image I_x in visual space can be defined as (1) at the bottom of the page, where $I_i, i = 1, \ldots, K$ are the K nearest neighbors of I_x in the visual feature space.

The basic idea of the proposed NNCS in visual space is to measure the semantic gap by using the low-level content features of image I_x to search for visually similar images first. The contextual (semantic) similarities of the images returned from the search are then calculated. If they share common contextual information, we can conclude these visually similar images also have very similar semantics and can thus be called content and context similar. This consistency shows that the low-level features of image I_x can express its semantic information well. Hence, the higher the NNCS is, the smaller the semantic gap would be.

In our experiment, we calculate the NNCS with $K = 500$[1] for all 2.4 million images and construct a large visual-textual confidence map. From this map, we can select candidate images with a high NNCS for later concept exploration and lexicon construction. The simplest way[2] to do this is to rank all images by their NNCS value and use a threshold to select the top N images. In our implementation, 36 231 top images are selected due to their relatively large size and memory concern with regards to the affinity propagation clustering algorithm described in Section II.

2) Clustering Using Affinity Propagation: After candidate images with small semantic gaps are selected, the next step is to cluster these images and extract the corresponding concept information. We use a very recently proposed affinity propagation method [17] for clustering because it is fast for large-scale data sets and requires no *prior* information (e.g., number of clusters).

Differing from traditional clustering methods, affinity propagation does not need to specify and fix the number of exemplars (representative centers). It starts with the construction of a similarity matrix. By viewing each data point as a node in a network, this method recursively transmits real-valued messages along the edges of the network until a good set of exemplars and corresponding clusters emerge. Affinity propagation has been successfully used in face image clustering, gene detection, sentence identification, etc. [17].

In order to cluster content and context similar images together, we define and construct a content-context similarity matrix based on content-context K-nearest neighbors. Intuitively, image I_j is the content-context K-nearest neighbor of image I_i only if I_j is both visually and textually nearest neighbors of image I_i. An example illustrating this is given in Fig. 4. I_i has $K(K = 5)$ visually and textually nearest neighbors $v_1,\ v_2,\ v_3,\ v_4,\ v_5$ and $t_1,\ t_2,\ t_3,\ t_4,\ t_5$. In this example, only image I_j and I_k are called the content-context neighbors of image I_i since they are both visual and textual nearest neighbors of image I_i. The content-context K-nearest neighbors of image I_i can be represented by $Content - Context\ kNN(I_i) = \{I_j,\ I_k\}$.

[1] K is chosen as a trade-off of both image coverage and computational complexity.

[2] We have tested with other selection methods, but simple thresholding gives quite reasonable candidate set and is very fast for implementation for large-scale image dataset.

$$NNCS_{visual}(I_x) = \frac{1}{K}\sum_{i=1}^{K} sim\_text(I_x, I_i) \text{ for } I_i \in Visual\_neighbors(I_x) \quad (1)$$

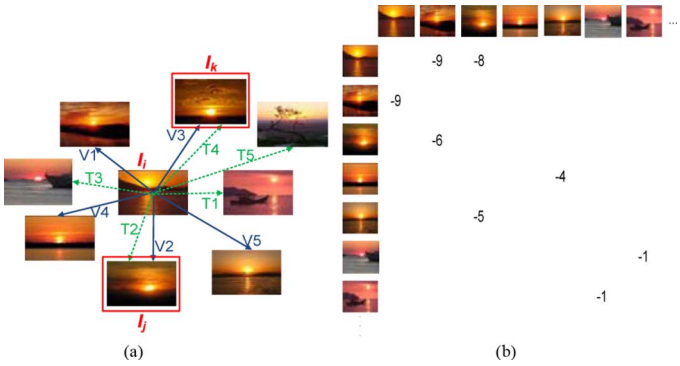


Fig. 4. (a) Content-context K-nearest neighbor. Images I_k and I_j are content-context K-nearest neighbors of image I_i since they are both visually and textually nearest neighbors of image I_i. (b) Content-context similarity matrix. Similarities between each of the two images are calculated and stored in a content-context similarity matrix.

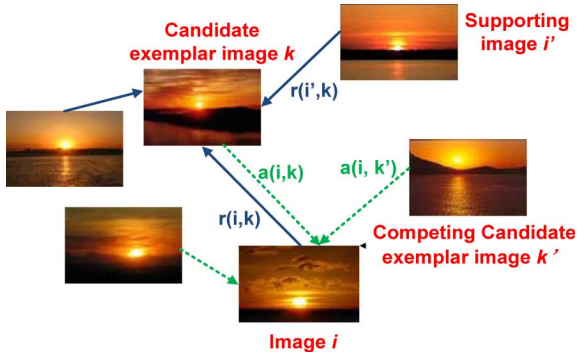


Fig. 5. Message passing between images. "responsibility" $r(i,k)$ and "availability" $a(i,k)$ messages are exchanged between images i and k to reflect the accumulated evidence of how well-suited image k is to be the exemplar for image I and the evidence of how appropriate it is for image i to choose point k as its exemplar, respectively.

Based on content-context K-nearest neighbors, we construct a 36 231 × 36 231 content-context similarity matrix P as shown in (2) at the bottom of the page (Fig. 4), where

$$\begin{aligned} sim(I_i, I_j) &= (1-\lambda)\cdot sim\_visual(I_i, I_j) \\ &\quad + \lambda \cdot sim\_text(I_i, I_j) \\ &= -(1-\lambda)\cdot dist\_visual(I_i, I_j) \\ &\quad - \lambda \cdot dist\_text(I_i, I_j). \end{aligned} \tag{3}$$

Matrix P describes the similarity of visual content and textual context between any two images I_i and I_j, $i, j = 1, 2, \ldots, 36\,231$. When I_j is the content-context neighbor of I_i, P_{ij} is set to their content and context similarity, which equals to the summation of the negative Euclidean distance of their visual features and the cosine distance of their textual features. Otherwise, P_{ij} is set to $-\infty$.

P_{ij} indicates how well an image I_i is suited to be the exemplar for image I_j. We assume that all images are equally considered to be exemplars. Hence, the preferences [17] are set to a common value—the median of P_{ij}. It should be noted that P_{ij} is not necessarily equal to P_{ji}. Fortunately, affinity propagation can be applied to a non-symmetric similarity matrix.

Affinity propagation is a message-passing algorithm. Two kinds of messages: "responsibility" $r(i,k)$ and "availability" $a(i,k)$ are exchanged between images i and k.

Shown in Fig. 5, the "responsibility" $r(i,k)$, sent from image i to the candidate exemplar image k, reflects the accumulated evidence of how well-suited image k is to be the exemplar for image i, taking into consideration other potential exemplars. The "availability" $a(i,k)$, sent from candidate exemplar image k to point i, reflects the evidence of how appropriate it is for image i to choose point k as its exemplar, considering the support from other images that image k should be an exemplar.

At the beginning, the availabilities are initialized to zero: $a(i,k) = 0$. Then, the responsibilities and availabilities are updated iteratively using the two rules in (4) and (5) at the bottom of the page, which let all candidate exemplars compete for ownership of an image and gather evidence from images to support each candidate exemplar.

After a fixed number of iterations, a good set of exemplars and corresponding clusters emerge. For image i, the image k that maximizes $a(i,k)+r(i,k)$ will be identified as its exemplar. If $k = i$, image i itself is an exemplar. The corresponding clusters are constructed by connecting each image to the exemplar that best represents it.

$$P_{ij} = \begin{cases} sim(I_i,\ I_j) & \text{for } I_j \in Content - Context\ KNN(I_i) \\ -\infty & \text{for } I_j \notin Content - Context\ KNN(I_i) \end{cases} \tag{2}$$

$$r(i,k) \leftarrow sim(i,k) - \max_{k'\,s.t.\,k'\neq k}\left\{a\left(i,k'\right) + sim\left(i,k'\right)\right\} \tag{4}$$

$$a\left(i,k\right) \leftarrow \begin{cases} \min\left\{0, r\left(k,k\right) + \sum\limits_{i'\,s.t.\,i'\notin\{i,k\}} \max\{0, r\left(i',k\right)\}\right\} & i \neq k \\ \sum\limits_{i'\,s.t.\,i'\neq k} \max\left\{0, r\left(i',k\right)\right\} & i = k \end{cases} \tag{5}$$

TABLE I
TOP 50 KEYWORDS IN THE IDENTIFIED LEXICON

Category	Concepts
Scene/Landscape	sunset , sky, beach, garden, lake, sunflow, water , firework, cloud, moon, sunrise, mountain, city, river, snow, rain, home, island
Object	flower, rose, butterfly, tree, bee, candle, bridge, leaf, eye, tulip, orchid, house, peacock, window, glass, bird, rock
Color	blue, red, yellow, green, pink, purple, orange, dark, golden
Season	fall, spring, summer, autumn
Others	small, wild

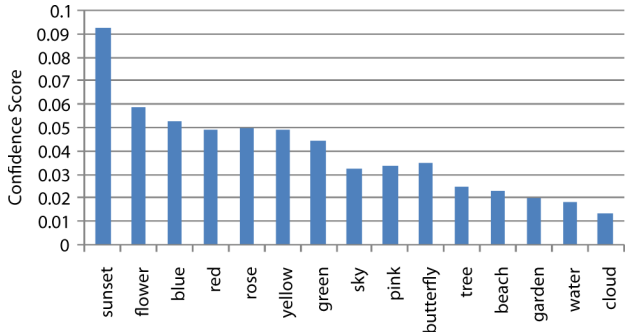


Fig. 6. Distribution of average confidence value. x-axis represents top 15 keywords (from left to right) in developed lexicon.

C. Text-Based Keyword Extraction

After candidate images are well clustered, a text-based keyword extraction algorithm is proposed to extract keyword (concept) information from these clusters.

Given a cluster C_i in the cluster pool C, the text-based keyword extraction algorithm is to find the most representative keywords by ranking all related keywords in this cluster. The related keywords are those that appear in the title or surrounding descriptions of images belonging to C_i. To be specific, the set of the related keywords of cluster C_i is denoted as W_i, and the relevance score of a keyword k_j to cluster C_i is denoted as $Score\_r(k_j, C_i)$.

Many different strategies could be applied to calculate $Score\_r(k_j, C_i)$. In paper [11], they show that an *if-ikf* strategy (image frequency-inverse keyword frequency) performs well when it finds keywords from surrounding texts of an image. Enlightened by the *if-ikf* strategy, we extend this strategy from image to cluster, defined as follows:

$$Score\_r(k_j, C_i) = \begin{cases} \frac{occurence(k_j, C_i)}{\ln(|W_i|+1)} & otherwise \\ 0 & W_i = \Phi \ or \ k_j \notin W_i \end{cases} \tag{6}$$

where $occurence(k_j, C_i)$ denotes the count of keyword k_j found in the title or descriptions of images belonging to cluster C_i.

Similarly, for the whole cluster pool C, we denote W as the combination of all W_i, i.e., $W = \bigcup_i W_i$. The relevance score of each keyword k_j in W to the whole cluster pool C can be denoted as $Score(k_j)$. It is the summation of the relevance scores of k_j to each $C_j, C_i \in C$. $Score(k_j) = \sum_{C_i \in C} Score(k_j, C_i)$. The assumption is that if k_j is a representative word for many clusters, it would also be an important keyword for the whole cluster pool.

Once the relevance score $Score(k_j)$ of each keyword k_j in W is obtained, we can select the top ones to make the final concept lexicon. The above concept lexicon selection procedure uses the NNCS in visual space to measure content (visually) similar images' contextual consistency in textual space. Hence, we name it the visual-central nearest neighbor confidence score algorithm (Visual-NNCS).

Due to limited space, we only list the top 50 keywords in Table I grouped in five categories: scene, object, color, season, and others.

IV. EXPERIMENTS

It is not a trivial task to evaluate the identified lexicon of high-level concepts with small semantic gaps. We design two experiments to evaluate the validity of the lexicon. First, we construct a confidence map for some concepts selected from the lexicon and compare their average confidence score. Secondly, we apply this developed lexicon on image annotation refinement. The superior performance demonstrates that this concept lexicon provides a reliable and effective list of concepts with small semantic gaps.

A. Confidence Map

One way to evaluate the lexicon is to calculate the confidence score of images labeled with the keywords in the list. Intuitively, the images labeled with the top keywords should have higher confidence values than images labeled with lower ranked keywords.

Hence, in our first experiment, we select the top 15 keywords from the lexicon. For each keyword w, we randomly selected 500 titled photos with this keyword from our 2.4 million web images database. It is our assumption that the image is labeled with the keyword if the keyword appears in its title. Then we compute the average NNCS of photos labeled with the same keyword.

The distribution of the average confidence score for each keyword is shown in Fig. 6. It can be seen that the confidence value decreases similarly to the keyword rank's depreciation. This figure clearly demonstrates that the images labeled with the top keywords have higher confidence value.

B. Image Annotation Refinement

As mentioned in Section I, the developed lexicon can be applied to help refine and re-rank annotated keywords. This is called *image annotation refinement*.

In this section, we apply the lexicon on the University of Washington (UW) dataset to refine the annotation results obtained by the search-based image annotation algorithm [11]. Two different refinement strategies are shown in Fig. 7.

UW dataset is a popular content-based image retrieval database, which has been used in many cited works and is downloadable from http://www.cs.washington.edu/ research/image-database/groundtruth/. For each image, there are about five manually labeled ground truth annotations. In total, it contains 1109

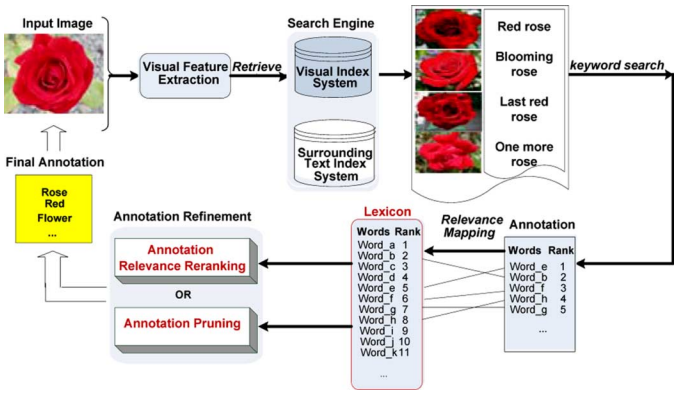


Fig. 7. Image annotation refinement scenario. First, a search-based image annotation algorithm is applied to annotate a given image. Then, the developed lexicon is applied to refine the annotations by re-ranking the candidate annotations with two refinement strategies: annotation relevance re-ranking and annotation pruning.

images and more than 350 unique words. In our evaluation, we strictly use the annotations of UW as the ground truth and all synonyms and non-appearing annotations are assumed to be incorrect.

Search-based image annotation is a very recent annotation algorithm [11]. It first uses the 64 color features to retrieve visually similar images. Next, it applies a keyword search to obtain a ranked list of candidate annotations from the surrounding texts of the retrieved images.

Given a query image, search-based image annotation is first employed to obtain a set of candidate annotations. Then, it uses the developed lexicon to refine the annotations by re-ranking the candidate annotations and reserving the top annotations (Fig. 7). In our experiments, two new refinement strategies are used: *annotation relevance re-ranking* and *annotation pruning*.

1) Annotation Relevance Re-Ranking: For each keyword appearing in the annotations of an image, for example $word\_a$, its relevance score, which reflects the relevance between the image and keyword, can be calculated as

$$Score\_r(word\_a) = \frac{1}{\ln(1+i)} \tag{7}$$

where i is the rank of $word\_a$ in the annotation of the image. The assumption is that the keyword with top rank is more important to the image and thus has a larger relevance score.

Similarly, in the constructed lexicon, a static score of each keyword could also be defined to reflect its appropriateness as an effective annotation. It should be noted that our word lexicon is constructed over the 2.4 million web images data set that we construct, which is independent from the UW dataset. Therefore, the static score is independent of the target images:

$$Score\_s(word\_a) = \begin{cases} \frac{1}{\ln(1+j)}, & word\_a \in LCSS \\ 0, & word\_a \notin LCSS \end{cases} \tag{8}$$

where j is the rank of the $word\_a$ in the lexicon list. If $word\_a$ does not appear in the lexicon list, its static score is defined as 0.

The final score of the keyword could be calculated as a weighted combination of the relevance score and the static score as follows:

$$\begin{aligned} Score\_c(word\_a) &= Score\_r(word\_a) \\ &\quad + \alpha \cdot Score\_s(word\_a) \\ &= \frac{1}{\ln(1+i)} + \alpha \cdot \frac{1}{\ln(1+j)}. \end{aligned} \tag{9}$$

In our experiments, α is set to 10 empirically from various tests.

From formula (9), we can see that if one keyword is ranked high in the annotation but appears low in the lexicon, its final rank will decrease. Similarly, if a lower ranked keyword is within the top keywords of the lexicon, it will be ranked higher in the final annotation.

2) Annotation Pruning: Annotation pruning is an alternative to annotation relevance re-ranking. The difference from annotation relevance re-ranking is that irrelevant annotations that do not appear in the lexicon are pruned. The basic assumption is that highly correlated annotations should be reserved and non-correlated annotations should be removed.

Since the original UW ground truth annotations include both keywords and phrases, we define two evaluation levels in our experiments to evaluate the annotation performance: *phrase*-level and *term*-level. In the phrase-level, an annotation is considered to be correct if and only if it is a ground truth annotation of the target image. In the term-level, both the ground truth annotation phrases and the result annotation phrases are divided into separate words. If a word appears more than once in the annotations of an image, only one is reserved. An annotated keyword is considered to be correct if and only if it appears in the ground truth annotation of the target image. In all of our experiments, we use exact string matching. The *Precision* and *Recall* are defined as shown in (10) and (11) at the bottom of the page, where n is the total number of test images. In our experiments, the number of the search-based annotation algorithm's annotation keywords is restricted to be no more than ten.

$$precision = \frac{1}{n}\sum_{k=1}^{n} \frac{\text{number of correctly annotated phrases (terms) in image } I_k}{\text{number of annotated phrases (terms) in image } I_k} \tag{10}$$

$$recall = \frac{1}{n}\sum_{k=1}^{n} \frac{\text{number of correctly annotated phrases (terms) in image } I_k}{\text{number of ground truth phrases (terms) in image } I_k} \tag{11}$$

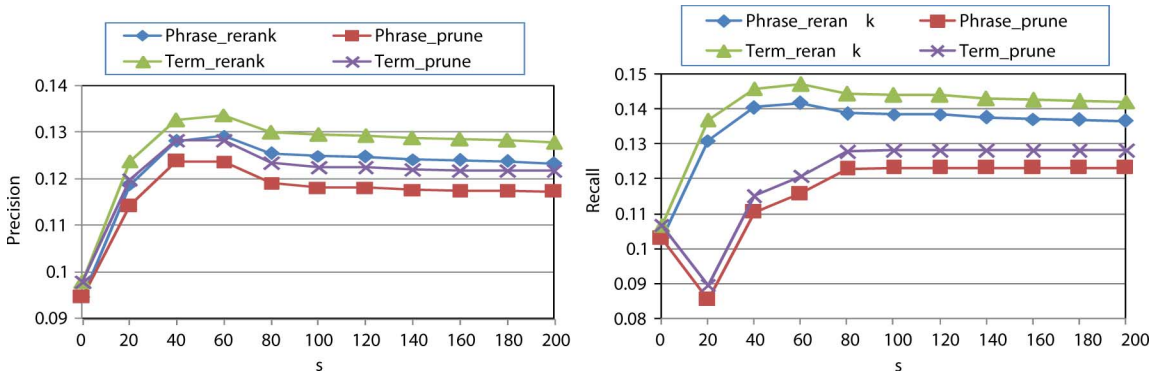


Fig. 8. Annotation precision and recall of different sizes of lexicon. Different sizes of lexicon-s are tested on annotation refinement. When s becomes larger, the refinement distinctively improves the original annotation's *Precision* and *Recall*. When s equals to or is larger than 100, *Precision* and *Recall* keep stable.

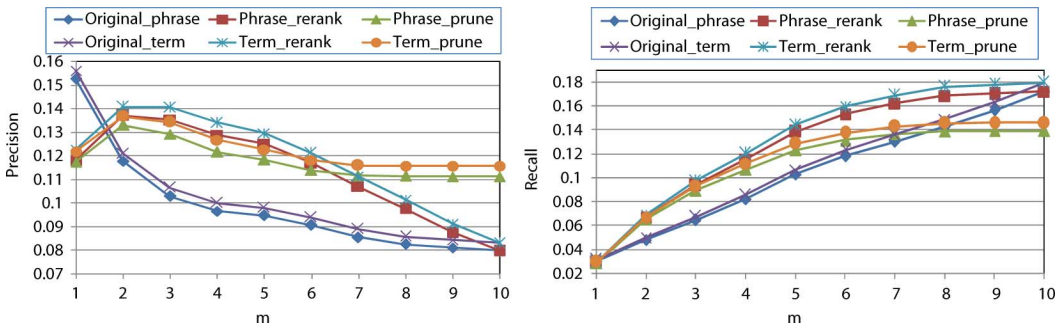


Fig. 9. Annotation precision and recall of different sizes of m. Different number of annotation results-m are tested on annotation refinement. When m ranges from 3 to 7, the *Precision* and *Recall* of refined annotation (annotation re-rank and pruning) are improved most.

3) Size of Lexicon: Obviously, the size of the lexicon is a crucial parameter in the annotation refinement process. It decides how many keywords will be used to refine or prune the final annotation. Let us denote it by s. To facilitate further evaluations, s is first determined by comparing the annotation performance at different values of s. The number of annotation results m is fixed to five since there are about five manually labeled ground truth annotations in the UW dataset. The *Precision* and *Recall* are shown in Fig. 8, with s changed from 0 to 200. When s is set to 0, it denotes the original annotation without refinement.

From Fig. 8, we find that *Precision* and *Recall* exhibit similar varying characteristics. The refinement distinctively improves the original annotation's precision and recall when s becomes larger. With the small lexicon size (e.g., $s = 20$), the annotated keywords of UW dataset may not appear in the top 20 concepts of generated lexicon from 2.4 million online photos. Hence with the pruning scheme, the recall rate could drop. As the number of concepts increases, the coverage of these concepts is enlarging. Therefore, the recall rate increases. The performance of refinement remains stable when s is equal to or greater than 100, which means that most annotation words of the UW dataset fall into the first 100 keywords in the lexicon. Therefore, s is set to be 100 in the following evaluations.

4) Size of Annotation: In Fig. 8, we can see that annotation refinement has consistently improved the performance when m is 5. In order to test the effect of different m values, m is varied in our second experiment from 1 to 10. The corresponding *Precision* and *Recall* are shown in Fig. 9, respectively. Three observations can be drawn from the results. First, when m ranges from 3 to 7, the *Precision* and *Recall* of refined annotation (annotation re-rank and pruning) are improved the most. When m reaches 10, the *Precision* and *Recall* of annotation re-rank become the same as the unrefined one since all annotation words are being counted for both methods. In addition, the *Precision* and *Recall* of annotation pruning remains the same, especially when m is larger than 7. It means that most of the top 7 annotation words produced by the search-based image annotation algorithm fall into the lexicon. Secondly, the absolute values of term-level evaluation are better than that of phrase-level, because search-based image annotation is a term-based annotation algorithm. Thirdly, the performance of annotation re-rank is better than pruning because the pruning method filters out some possible correct annotations.

5) Comparison With Other Lexica: In our last experiment, we studied three different lexica of semantic concepts, where each set is larger than the previous one.

LCSS: Our developed lexicon list of concepts with small semantic gaps. To be consistent with previous experiments, we still use the top 100 keywords for annotation refinement.

LSCOM: large-scale concepts ontology for multimedia [13], a standardized lexicon established on broadcast news video from TRECVID benchmark. The largest word list, which can be downloaded from http://www.ee.columbia.edu/ln/dvmm/lscom/, contains 858 terms.

WordNet: A very large lexical database of English words. Nouns, verbs, adjectives, and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct

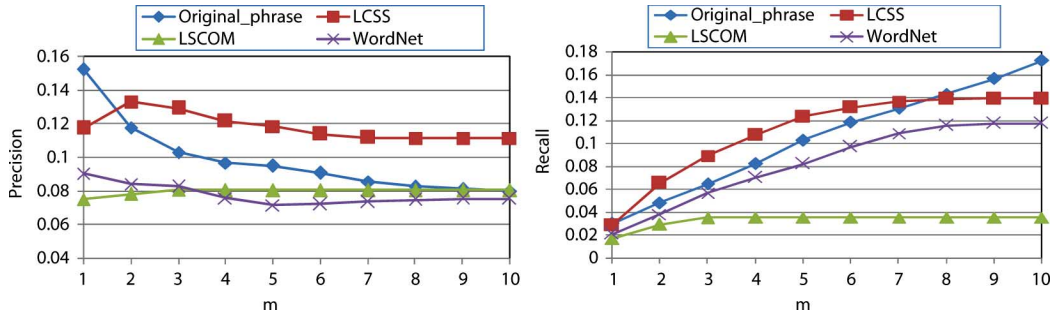


Fig. 10. Annotation precision of different lexica. Three different lexica of semantic concepts, LCSS, LSCOM, and WordNet, are applied for annotation refinement. LSCOM and WordNet cannot improve the annotation precision and recall. Our constructed LCSS outperforms the other two lexica and is more effective for image annotation refinement.

concept. Because most image annotation words are nouns and adjectives, for this study, we use a total of 100 303 nouns and adjectives terms from WordNet 2.1 [18].

Since LSCOM and WordNet do not have word rank, in this experiment, we only compare their annotation pruning performance on the annotations of the UW dataset. From Fig. 10, it is clear that LSCOM and WordNet do not improve the annotation precision at all and perform much worse than LCSS. The reason is that many correct annotations are not included in these two lexica, and thus are pruned. It validates that these two lexica are not good concept corpora with small semantic gap for web-scale image annotations. Instead, LCSS is demonstrated to be more effective for image annotation refinement.

V. Lexica Family for Concept-Based Image Retrieval

In the above sections, we proposed a novel way to construct a lexicon of high-level concepts with small semantic gaps. These concepts were extracted from the textual information of candidate images by measuring their consistency in visual feature space and semantic textual space.[3] In fact, a single lexicon is not enough for various types of concepts in large-scale image data. One important reason is that in different visual spaces, images of concepts distribute differently. Two concepts may be far away from each other in one visual feature space, but might be much closer to each other in another space. For example, "wood" and "sand" have similar color features, but they are totally different in the texture feature space. Thus, a lexicon based on a single visual feature is insufficient for presenting concepts. It is necessary to construct lexica based on different low-level features. A family of feature-based lexica can provide appropriate options for feature selection of specific concepts.

A. Feature-Based Lexica

In this section, we extract three low-level features for 2.4 million web images, as shown in Table II. Given a specific low-level feature (color feature or texture feature in Table II), using the Visual-NNCS algorithm, we extract a color-based lexicon (top 100 concepts) by using the color feature and develop a texture-based lexicon (top 100 concepts) by using the texture feature. We then compare the color-based lexicon and the texture-based lexicon as shown in Fig. 11.

TABLE II
Three Low-Level Features of Image Dataset

Low-level features	Dimension	Descriptions
Color	50	6-dim color moment(LUV) and 44-dim banded auto-color correlogram (HSV)
Co-occurrence Texture (COT)	16	16-dim normalized vector as measurement of global grey-level co-occurrence matrix
Wavelet Texture (WT)	128	128-dim vector of wavelet parameters

Fig. 11. Concepts in feature-based lexica. The color-based lexicon contains 65 concepts, shown in the box with the solid line. There are 81 concepts that belong to the texture-based lexicon, which are marked with the dotted line. The two lexica share 42 concepts (within the overlapping area), which have inherently small semantic gaps based on either the color feature or the texture feature.

After removing some noisy concepts, there are a total of 104 meaningful concepts in the two lexica (Fig. 11). The color-based lexicon contains 65 concepts shown in the box with the solid line. There are 81 concepts that belong to the texture-based lexicon, which are marked with dotted line. The two lexica share 42 concepts (within the overlapping area), which have inherently small semantic gaps based on either the color feature or the texture feature. For example, "sunset" is such a concept. By using the color feature or the texture feature, we can always model a classifier for it with good performance and get relatively correct annotations. Some concepts have small semantic gaps based only on one specific low-level feature. Some botanic

[3]The early work was published in CVPR'08 [10]. In our extended work, we propose the textual-central nearest neighbor score to calculate semantic gaps. We construct a lexicon family of high-level concepts with small semantic gaps (LCSS) with different visual features (feature-based lexicon) and different content-contextual consistency (consistency-based lexicon). LCSS is very useful for a search engine to choose appropriate low-level features and better search methodology to retrieve a specific concept.

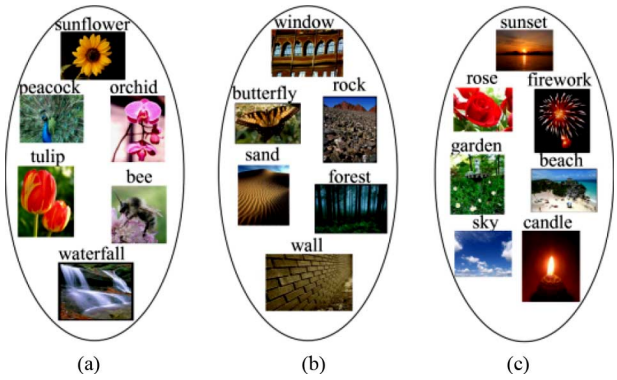


Fig. 12. Examples of feature selection for concepts. (a) Color feature is the best choice for concepts, e.g., sunflower, peacock, and orchid. (b) Texture feature is the best choice for concepts, e.g., window, butterfly, and rock. (c) Combined color and texture feature is the best choice for concepts, e.g., sunset, rose, and firework.

concepts such as sunflower, bud, leaf, bloom, and animal concepts such as peacock, bee, and spider have small semantic gaps only in the color feature space. However, some scene-level concepts—storm, dawn, fog, snow, and sand—and landscape concepts like river, harbor, forest, and hill have small semantic gaps only in texture feature space.

By comparing different lexica, we can select a more appropriate low-level feature as a representative feature, that is, feature selection for concepts. For those concepts that have a small semantic gap only based on the color feature, like sunflower, peacock, and orchid [as shown in Fig. 12(a)], the color feature is the best choice for feature selection. For those concepts which have a small semantic gap only based on the texture feature, like window, butterfly, and rock [as shown in Fig. 12(b)], the texture feature is the best choice. For some concepts that have a small semantic gap based on either the color feature or the texture feature, such as sunset, rose, and firework [as shown in Fig. 12(c)], we can choose to combine color and texture as an appropriate representation. Some other examples can be found in Fig. 12.

B. Consistency-Based Lexica

The second reason for the insufficiency of a single lexicon is that semantic gaps not only depend on features, but also depend on the content-contextual consistency measurement. In Section III, formula (1), we define the NNCS in visual space to measure content (visually) similar images' contextual consistency in textual space. Hence, we name it the Visual-NNCS. This Visual-NNCS considers visually similar images' consistency in textual space but ignores textually similar images' consistency in visual space.

In fact, textually similar images could be dissimilar in visual space [Fig. 3(c)]. The concepts' consistency should be measured by considering both visual consistency and textual consistency, and maybe more. Therefore, we propose another textual-central nearest neighbor confidence score (Textual-NNCS) to measure contextual similar image's content consistency.

Textual-NNCS: for a given image I_x, first find its K neighbors in textual space $\{I_i | i = 1, 2, \ldots, K\}$, then calculate the average of visual similarity between I_x and I_i, shown in (12) at the bottom of the page. $sim\_visual(I_x, I_i)$ can be calculated by measuring the negative visual features' Euclidean distance between I_x and I_i. The proposed Textual-NNCS measures the semantic gap by using the high-level context (semantic) of image I_x to search most contextually similar images first. Next, their visual similarities are calculated. If they share common visual information, we can conclude that these semantically similar images also have very similar visual information. Thus, they are called context and content similar. This consistency shows that the high-level semantics of image I_x can express its low-level visual information well. Hence, the higher the Textual-NNCS is, the smaller the semantic gap would be and the tighter the visual consistency the concept concerned in this image has.

The definitions of the two different NNCS algorithms reflect different search methods: content-based search (visual-based) and context-based (textual-based) search. Lexica with small semantic gaps based on different NNCS algorithms can provide suggestions for search methods selection. Therefore, we use a 50-dim color feature as the visual representation of the images and construct two lexica by using the Visual-NNCS and Textual-NNCS algorithms, respectively. The two different lexica are obtained by calculating the NNCS for each image, selecting candidate images, clustering, and extracting concepts. Similarities and differences between these two lexica are shown in Table III.

In Part I of Table III, there are seven categories of concepts with small semantic gaps only based on Visual-NNCS. Hence, for concepts like firework and rose, content-based search is more preferable over context-based search. By contrast, the five categories of concepts within Part II of Table III have small semantic gaps only based on Textual-NNCS. For concepts like girl and street, it is better to search them based on textual information than visual information. In addition, Part III of Table III consists of concepts which have small semantic gaps based on either Visual-NNCS or Textual-NNCS. For these concepts such as sunset, both content-based search and context-based search have good performance. The above experimental results are very useful for choosing search methods for different concepts.

VI. Conclusion and Future Work

In this paper, we have presented an innovative framework for automatically constructing a family of lexica with small semantic gaps from a large web-based image dataset.

$$NNCS_{textual}(I_x) = \frac{1}{K}\sum_{i=1}^{K} sim\_visual(I_x, I_i)\ for\ I_i \in Textual\_neighbors\ (I_x) \tag{12}$$

TABLE III
COMPARISON OF TWO LEXICA BASED ON TWO NNCS ALGORITHMS

Part I: Visual-central NNCS	
Category	**Concepts**
Scene	firework, sunrise, rain, wild
Landscape	bay, field, home, house, coast, ocean, pier, hill
Color	yellow, green, pink, purple, orange, golden
Object	candle, moon, drop, boat, saw
Plant	rose, sunflower, orchid, tulip, daisy, lily, irid, leaf, bloom, glass
Animal	bee, peacock, fish, bird
Season	spring, summer, autumn
Part II: Textual-central NNCS	
Category	**Concepts**
People	girl, man, woman, model, angel, nude, sister, children, male, face
Animal	cat, tiger, dog, wolf
Water	creek, valley, canyon, stream
Place	street, road, church, castle, cemetery, market, square, metro, studio, village, town
object	stone, chain, crater
Part III: Either of Visual-central NNCS and Textual-central NNCS	
Category	**Concepts**
Scene	sunset, sky, shadow, city, water snow, storm, ice, cloud
Landscape	fall, lake, river, garden, beach, mountain, bridge, waterfall, island
Color	red, blue, dark
Object	eye, rock, key, window, flower, tree

Our major contributions are: 1) This work sheds some light in answering the question "what specific concepts have small semantic gaps?" Among the hundreds or even thousands of multimedia concepts, this work is the first of its kind which tries to derive which semantic concepts should be focused on first in data collection and modeling. 2) It also provides a candidate pool of good semantic concepts to annotate other image datasets. Thus, it can be used for annotation refinement and rejection. 3) This lexica family also contains feature-based lexica and consistency-based lexica. Feature-based lexica provide feature selections for image retrieval with concepts. Consistency-based lexica obtained by measuring semantic gap from both visual and textual space can provide good suggestions for choosing a search model for concepts. The lexica family will provide more options to different modeling methods given specific features and has many potential applications in concept detection, query optimization, and multimedia information retrieval.

It should also be noted that these concepts are related to low-level visual features. In addition to the current color and texture features used in this work, future work will investigate shape features, SIFT features, and some local features in order to construct more comprehensive feature-based lexica. Although a number of semantic concepts have been developed for multimedia information retrieval, some questions still remain, e.g., how many semantic concepts are necessary [19]? Our work is the first step in answering such questions. More systematic approaches to modeling semantic gaps are still worth investigating.

REFERENCES

[1] S. Agarwal and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1475–1490, Nov. 2004.

[2] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories," in *Proc. IEEE Conf. Computer Vision Pattern Recognition Workshop on Generative-Model Based Vision*, 2004, p. 178.

[3] G. Griffin, A. Holub, and P. Perona, Caltech-256 Object Category Dataset, Caltech Tech. Rep., 2007.

[4] [Online]. Available: http://www.pascal-network.org/challenges/VOC.

[5] J. Li and J. Z. Wang, "Real-time computerized annotation of pictures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 985–1002, Jun. 2008.

[6] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos, "Supervised learning of semantic classes for image annotation and retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 394–410, Mar. 2006.

[7] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proc. SIGCHI Conf. Human Factors in Computing System*, 2004, pp. 319–326.

[8] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: a database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, no. 1–3, pp. 157–173, 2008.

[9] X. J. Wang, L. Zhang, F. Jing, and W. Y. Ma, "Annotating images by mining image search results," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1919–1932, Nov. 2008.

[10] Y. Lu, L. Zhang, Q. Tian, and W. Y. Ma, "What are the high-level concepts with small semantic gaps?," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[11] C. Wang, F. Jing, L. Zhang, and H. J. Zhang, "Scalable search-based image annotation of personal images," *Multimedia Syst.*, vol. 14, no. 4, pp. 205–220, 2008.

[12] J. Ponce *et al.*, "Dataset issue in object recognition," in *Toward Category-Level Object Recognition*. New York: Springer, 2006, pp. 29–48.

[13] M. R. Naphade *et al.*, "Large-scale concept ontology for multimedia," *IEEE Multimedia*, vol. 13, no. 3, pp. 86–91, Jul.–Sep. 2006.

[14] C. G. M. Snoek, M. Worring, J. C. Van Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proc. ACM Multimedia*, 2006, pp. 421–430.

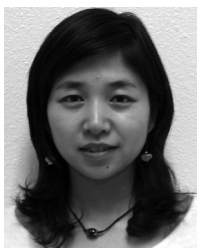
[15] A. Hauptmann, R. Yan, W. H. Lin, M. Christel, and H. Wactlar, "Can high-level concepts fill the semantic gap in video retrieval? A case study with broadcast news," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 958–966, Aug. 2007.

[16] L. Zhang, Y. Hu, M. Li, W. Y. Ma, and H. Zhang, "Efficient propagation for face annotation in family albums," in *Proc. ACM Multimedia*, 2004, pp. 716–723.

[17] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976, 2007.

[18] G. A. Miller, "Wordnet: A lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

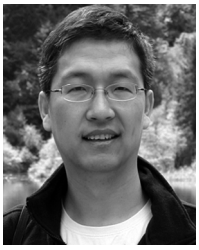
[19] A. Hauptmann, R. Yan, and W. H. Lin, "How many high-level concepts will fill the semantic gap in video retrieval?," in *Proc. ACM Int. Conf. Image and Video Retrieval*, 2007, pp. 627–634.



Yijuan Lu (M'05) received the Ph.D. degree in computer science from the University of Texas at San Antonio in 2008.

She is an Assistant Professor in the Department of Computer Science, Texas State University, San Marcos, TX. During 2006–2008, she was a summer Intern Researcher at FXPAL lab, Web Search & Mining Group, Microsoft Research Asia (MSRA), National Resource for Biomedical Supercomputing (NRBSC) at the Pittsburgh Supercomputing Center (PSC), Pittsburgh. She was an Intern Researcher at Media Technologies Lab, Hewlett-Packard Laboratories (HP) 2008, and a research fellow of Multimodal Information Access and Synthesis (MIAS) Center at the University of Illinois at Urbana-Champaign (UIUC) in 2007. Her current research interests include multimedia information retrieval, computer vision, machine learning, data mining, and bioinformatics. She has published extensively and serves as a reviewer for top conferences and journals.

Dr. Lu is the 2007 Best Paper Candidate in the Retrieval Track of Pacific-Rim Conference on Multimedia (PCM) and the recipient of the 2007 Prestigious HEB Dissertation Fellowship, 2007 Star of Tomorrow Internship Program of MSRA. She is a member of ACM.



Lei Zhang (M'04) received the B.S. and M.S. degrees in computer science from Tsinghua University, Beijing, China, in 1993 and 1995, respectively. After two years working in industry, he later returned to Tsinghua University and received the Ph.D. degree in computer science in 2001.

He is a lead researcher in the Web Search and Mining Group at Microsoft Research Asia, Beijing, and an Adjunct Professor of Tianjin University, Tianjin, China. He currently directs a team pursuing new research directions on social media search. Team projects include multimedia content analysis, web-scale image annotation, and information mining for travel search. He is the author or coauthor of more than 80 published papers in fields such as content-based image retrieval, computer vision, web search, and information retrieval. He also holds 11 U.S. patents for his innovation in face-detection, red-eye reduction, and image retrieval technologies.

Dr. Zhang is an ACM member, has served as program co-chair of MMM 2010, and has served on international conference program committees, including ACM Multimedia, WWW, SIGIR, WSDM, ICME, MMM, etc.

Jiemin Liu received the B.S. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007, where she is pursuing the Ph.D. degree in the Department of Electronic Engineering.

From 2007 to 2010, she was a Master student in Shanghai Jiao Tong University and Microsoft Research Asia (MSRA) Joint-Education Program. During 2007–2009, she was an Intern Researcher at Internet Media Group and an Intern Engineer at Search Technology Center, MSRA. Her current research interests include multimedia, computer vision, data mining, and machine learning.

Qi Tian (SM'03) received the B.E. degree from Tsinghua University, Beijing, China, in 1992 and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC) in 2002.

He is currently an Associate Professor in the Department of Computer Science, the University of Texas at San Antonio (UTSA). He took one-year Faculty Leave at Microsoft Research Asia (MSRA), Beijing, during 2008–2009 in the Internet Media Group. He was a Visiting Scholar at UIUC MIAS center (2007), a Visiting Researcher at MSRA (summer 2007), a Visiting Professor in NEC Laboratories America, Inc. (summer 2003), and a Visiting Researcher (2001) in MERL, Cambridge, MA. He is currently an Adjunct Professor of Zhejiang University and Xidian University, China. His research interests include multimedia information retrieval and computer vision. He has published over 100 refereed book chapters, journal, and conference papers in these fields. His research projects were funded by ARO, DHS, HP Lab, SALSI, CIAS, and the Chinese Academy of Science.

Dr. Tian was the coauthor of a Best Student Paper in ICASSP 2006 and coauthor of a Best Paper Candidate in PCM 2007. He was nominated for the 2008 UTSA President Distinguished Research Award. He received the ACM Service Award in 2010 for ACM Multimedia 2009. He has been serving as a Program Chair and an Organization Committee Member for ACM Multimedia (2009), CIVR (2010), ACM ICIMCS (2009), ACM LSMMRM (2009), MMM (2010), VIP (2007, 2008), IMAI 2007, MIR (2005), and Session Chairs and TPC members for over 120 IEEE and ACM Conferences, including ACM Multimedia, SIGIR, ICCV, ICME, ICASSP, ICPR, MIR, VCIP, and PCM. He is a Guest Co-Editors for the IEEE TRANSACTIONS ON MULTIMEDIA, *ACM Transactions on Intelligent Systems and Technology* (ACM TIST), *Journal of Computer Vision and Image Understanding*, and *EURASIP Journal on Advances in Signal Processing* and is in the Editorial Board of the *Journal of Multimedia*. He has been a Member of ACM since 2004.