Universität Augsburg

Multimodal Ranking for Image Search on Community Databases

Fabian Richter, Stefan Romberg
Eva Hörster, Rainer Lienhart

Report 2011-07 März 2011

Institut für Informatik
D-86135 Augsburg

Multimodal Ranking for Image Search on Community Databases

Fabian Richter, Stefan Romberg, Eva Hörster, Rainer Lienhart

Multimedia Computing Lab
University of Augsburg
Augsburg, Germany
{richter, romberg, hoerster, lienhart}@informatik.uni-augsburg.de

ABSTRACT

Searching for relevant images given a query term is an important task in nowadays large-scale community databases. The image ranking approach presented in this work represents an image collection as a graph that is built using a multimodal similarity measure based on visual features and user tags. We perform a random walk on this graph to find the most common images. Further we discuss several scalability issues of the proposed approach and show how in this framework queries can be answered fast. Experimental results validate the effectiveness of the presented algorithm.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; I.4.10 [**Image Processing and Computer Vision**]: Image Representation—*multidimensional*

General Terms

Algorithms

Keywords

Image Ranking, Image Retrieval, PageRank, Graph

1. INTRODUCTION

With the emergence and spread of digital cameras in everyday use the number of images in personal and online collections grows daily. For example, the Flickr^{TM} photo repository now consists of more than four billion images. Such huge image databases require efficient techniques for navigating, labeling, and searching.

In this work we focus on the goal of selecting relevant images given a query term, i.e. finding images showing content that most people associate with the query term. More specifically we aim to solve this image search problem on a large-scale community database such as Flickr where images are often associated with different types of user generated metadata, e.g. tags, date & time, and location.

Our proposed image ranking approach has been inspired by [8] where the PageRank method [18] has been adapted to the visual domain. The PageRank approach is a method to rank webpages according to their importance. It builds a graph representing the link structure of the web. The importance of a webpage is assumed to be proportional to the number of hyperlinks pointing towards this page, i.e. the number of pages linking it. Transferring this approach to our image search task we assume that the relevance or importance of an image is proportional to the number of images showing similar content. As we consider community databases, i.e databases with images from many different authors/photographers, this assumption is justified by the following: If an image has many close neighbors all showing the same content and being associated with similar metadata then the respective images' authors agree that this is an important shot of the respective content.

The main difficulty in such an approach is to reasonably define the similarity between two images, i.e. to determine if two images show the same content. The authors in [8] calculate the images' distance based on the number of matching local features between two images. This approach works well for landmarks or product images as in such cases typically many images exist showing the exact same object. However, when searching for object categories or scenes we cannot expect to reliably match the local image descriptors. Thus we use a more sophisticated image description based on automatic content analysis. Moreover we do not rely solely on the automatically extracted visual content description for similarity definition, but we also exploit an image description based on the available metadata. More specifically we also use an representation based on the author's tags.

However, establishing links between all pairs in a huge image collection does not scale well, as this results into a complete graph and computing similarities between images is costly. Thus we also consider scalability issues when designing the link structure of our graph. The original PageRank method introduces only a limited amount of links in the graph due to the hyperlink structure of the web. Opposing to the hyperlink structure in the web context, there exists no similar link structure between images which we can exploit in our image graph. Therefore we rely on a nearest neighbor approach and compute similarities only between images in small subsets.

MIR'10, March 29–31, 2010, Philadelphia, Pennsylvania, USA.
Copyright 2010 ACM 978-1-60558-815-5/10/03 ...$10.00.

Finally, computing a separate relevance score for each image and each query term is computationally inefficient. Thus, based on our scalable nearest neighbor approach, we show how to compute the relevance of an image in a query-independent fashion. We evaluate our proposed image retrieval method extensively on a real-world large-scale database in user studies.

1.1 Related Work

There exist many works addressing the task of searching and ranking photos in (community) databases. For instance there are approaches that aim to find representative images of landmarks [25, 10, 2, 12]. Another work aims to find iconic object images [1] using cluster centroids and identifying images with clear foreground objects. There are image search approaches that rely on the user tags associated with the images. The main challenge is here posed by the ambiguity and subjectivity of user annotations in such community databases, making the direct application of text search approaches difficult. Therefore many approaches analyze and exploit the visual image content to improve the noisy labeling. Li et al. [11] and Liu et al. [15] both propose methods that use a visual content description to learn a tag's relevance to an image. In [11] the authors determine a tag's relevance by nearest neighbor voting. The latter work builds a graph using the tags associated with an image and performs a random walk to determine the tag relevance. The re-ranked tag lists are then used to retrieve images. We will use this approach as baseline for our user studies conducted in Section 4.

On the other hand there are approaches that directly analyze images and rely only on a visual content description, e.g. [4, 8], where the former work uses a classifier. However, a classifier needs to be trained on (carefully) labeled data which is not available in most scenarios.

Recently there have been some works exploiting multiple modalities for image search applications. Raguram and Lazebnik [21] perform joint clustering in a space built from visual and tag descriptors to find iconic summaries of abstract concepts. Wang and Forsyth [24] retrieve object images from webpages by analyzing the surrounding text and the image itself. In [23], Schroff et. al. use the surrounding text of web images for re-ranking purposes before training a SVM classifier based on visual features.

Our multimodal ranking approach has been inspired by the graph-based approach presented by Jing and Baluja [8]. Here the authors construct an image graph where vertices represent images and edge weights are proportional to the visual similarity between two images. An importance score is computed for each image and query term by performing random walk on this graph. However, in contrast to their product image search scenario, our goal is to perform retrieval of objects categories and scenes from community databases with very diverse images depicting objects in their natural context. Also, a query-dependent graph is used in [8] to compute the importance score. In this work we propose to compute a global ranking independent of any predefined query. We use multiple modalities to build our image graph, more precisely visual features and user tags. We show that our multimodal ranking method improves performance over the unimodal case. The work by Winston et al. [7] is similar to our method. However they employ a context graph in a different domain as they rank videos based on multimodal story-level similarities. Text transcriptions and visual similarity based on near-duplicate detection are used to build a graph which in turn is refined by random walk.

We also address scalability issues in our approach. In order to let our approach scale with the steadily growing size of image repositories, we exploit a nearest neighbor approach for graph construction. This idea has been motivated by [5], where the authors propose a framework for structural analysis of image databases using spectral clustering and spectral dimensionality reduction. The experimental results presented in [7] as well as the discussion in [19] prove the rationality of nearest neighbor approaches in a random walk context.

1.2 Contribution

The main contributions of this paper are:

1. We present a query-by-term image ranking approach that relies on a graph where the images are linked by their similarities. We show that using similarities based on both user annotations and visual content improves results.

2. Based on this image ranking approach we propose a self-contained image retrieval method which is designed to scale well with the increasing size of image repositories. More specifically we show how to appropriately modify the link structure of the graph and how to efficiently compute the image similarities needed to build the graph.

3. In our experiments on a large-scale real-world database we demonstrate the effectiveness of our approach. Our system yields highly satisfactory retrieval results for different kinds of query terms such as object and scene categories. In addition, we evaluate an extension such that the ranking score is computed independently of the query term resulting in a very effective, scalable image search.

1.3 Organization

This paper is organized as follows: Section 2 describes our proposed image ranking approach. In Section 3 we discuss the implementation of the presented method in more detail and address scalability issues. We evaluate the approach experimentally on a large-scale image database in Section 4. Section 5 summarizes our results.

2. APPROACH

Given a large-scale collection consisting of images and their respective metadata our goal is to find images relevant to a given query term. We define a relevant image as one showing content that most people associate with the query term. For now we focus on the case where only one query term is given, however our method can be extended to multi-term queries.

In order to perform image search given a query term, we start with a broad set of images that satisfy the query. In our implementation this set simply includes all images that have been tagged with the query term by their authors. Since this image set is derived automatically based on this simple constraint, it contains a significant number of noisy images not necessarily showing the desired image content due to

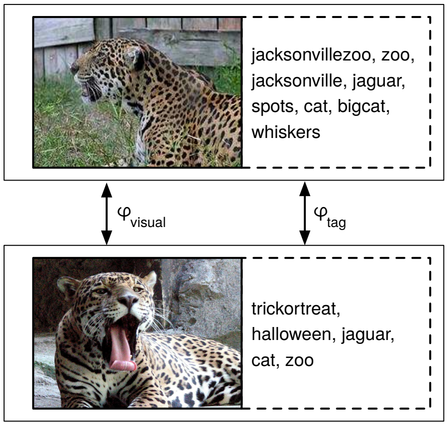
jacksonvillezoo, zoo, jacksonville, jaguar, spots, cat, bigcat, whiskers
φ_{visual}
φ_{tag}
trickortreat, halloween, jaguar, cat, zoo

Figure 1: Images are compared by using the similarities in two different domains, i.e. by using visual and textual features (tags).

the subjectivity and ambiguity of tags. Besides these images that are somehow related to the query term, we have neither additional data nor information about which images are preferred. Thus, we need to determine a score for each image indicating its relevance or importance to the current query.

Assuming that the importance of an image is proportional to the number of images showing similar content, we build a graph representing the relationships of all images in the database. Its vertices represent our images and the edges their multimodal similarity. Those multimodal similarities are based on visual and textual features. We then perform a random walk on this graph to determine a score for each image indicating its importance. This *importance score* reflects the likelihood of arriving in a certain vertex after a random walk over the given graph. We can then automatically rank the images in the former mentioned subset according to their importance score.

In the following subsections, we describe in detail how we build our similarity graph and review the random walk.

2.1 Multimodal Similarity Graph

The link structure as well as the weights associated with the edges are fundamental for the performance of our graph-based approach. Hence, the first step in building a similarity graph is to define an appropriate distance measure for comparing images.

Previous work [8] on image ranking uses only the visual content to compare images. On the other hand it has been shown in the context of query-by-example retrieval that image descriptions based on user annotations outperform a representation based on visual features. Moreover it has been shown in recent work [22, 13] that using multiple cues to find similar images boosts performance over using a single modality, either tags or visual features. Therefore we propose to use a distance measure for image comparison that combines the two modalities – user annotations and image content (see Figure 1).

We start by computing two descriptions for an image, one for each modality. In our implementation we represent our images by automatically extracted topic distributions. Details of our image representations are discussed in Section 3. It should be noted that our ranking system is not constrained to specific representations or modalities. However we use a low-dimensional vector description of an image in its respective modality as it allows to easily compute distances between two such vector representations.

Assuming we have a vector representation of the image content for each modality ω separately, one based on text features, i.e. tags, and the other based on visual features, we define an *image relevance score* as

$$\varphi_\omega(i,j) = \exp(-\frac{\|I_\omega(i) - I_\omega(j)\|_1}{\sigma_\omega}) \qquad (1)$$

where $\|I_\omega(i) - I_\omega(j)\|_1$ denotes in our case the L_1 distance of the representation $I_\omega(i)$ and $I_\omega(j)$ of images i and j. σ_ω is a normalization constant that was set equal to the median of the pairwise L_1 distance of all images. With the equation above we obtain a relevance score between each pair of images for each modality. We then fuse both scores linearly to a combined image similarity measure s_{ij}:

$$s_{ij} = \beta \cdot \varphi_{visual}(i,j) + (1-\beta) \cdot \varphi_{tag}(i,j) \qquad (2)$$

where $\beta \in [0,1]$ determines the weight between visual- and tag-based features. In Section 4 we evaluate the optimal setting for β experimentally by user studies.

Using the above defined similarity measure we are able to build our image graph. However this results in a complete graph (i.e. a graph having links between all images). Establishing, storing, and processing a fully connected graph becomes difficult when considering very large image databases, as the number of links would grow quadratically. Therefore we link each image only to its k-nearest neighbors, thus the number of edges grows only linearly in the number of images. Given an image we consider other images to be among the k-nearest neighbors if the distance between the corresponding vector representations is among the smallest k distances. This is equivalent to choosing the k images as neighbors which have the largest similarity according to Equation 2.

It should be noted that due to the nearest neighbor approach, the link structure of our graph differs considerably from a complete graph. Besides its resulting sparsity, links established in the graph are not bidirectional in general. Although the used similarity defined in Equation 2 is symmetric, the neighborhood of an image is not as it is defined by its k-nearest neighbors, not by the absolute distance value itself. Figure 2 visualizes the local structure of such an image graph. As we target the search in community databases where users upload their images, it is necessary to take special care to avoid artifacts introduced by users. For instance, a single user may contribute multiple images to our image graph. Each link from one image to another represents a vote for the other image's relevance. In order to limit a user's influence, we apply the following two restrictions:

- A user may not vote for any of his own images. That is, no links between images of the same user are allowed as this would make the ranking vulnerable to manipulation by a single user.

- If an image has an incoming link from more than one image of a particular owner, the respective link weights are normalized by the number of incoming links originating from that owner's images. Alternatively we could keep only the in-link from the best matching image.

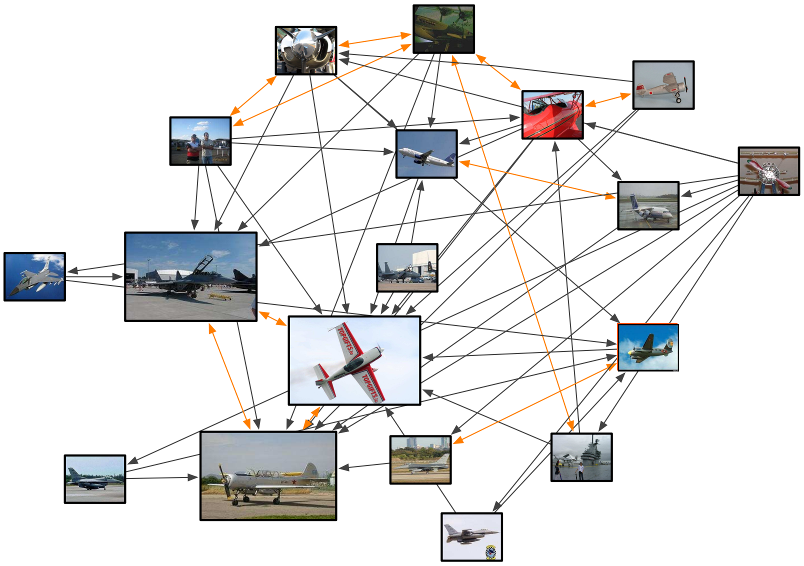

Figure 2: Example for the link structure established in the image graph according to multimodal similarities. Note that there are both unidirectional (gray) and bidirectional links (orange) due to our nearest neighbor approach.

The latter aspect is especially important since users tend to upload series of images which often show similar visual content and have the same or similar textual annotation. Thus, it is very likely that these near-duplicates share their nearest neighbors. Hence, an image voted by such a group or the group itself would be influenced overly strong by a single user.

2.2 Random Walk

Having determined the graph representing the relationship between the images in our database we now perform a random walk on this graph. The stationary distribution of the random walk process gives us a value for each image which we use as their respective relevance scores.

Let $\mathcal{G}$ denote the image graph. Each vertex of $\mathcal{G}$ corresponds to a certain image. The edges of the graph are established as discussed in the previous subsection. The random walk then traverses the graph according to its link structure, i.e. the probability of following an edge is given by its associated weight. The final computed random walk score for a vertex corresponds to the likelihood of arriving at this vertex after random walk over $\mathcal{G}$. Thus, in order to apply the iterative random walk algorithm to $\mathcal{G}$, we construct the corresponding transition matrix $\mathbf{P} \equiv [p_{ij}]_{n \times n}$, a row-stochastic matrix describing the transition probabilities used in the random walk process, i.e. $p_{ij} = p(j|i)$ is the probability of arriving at vertex j in one step given the current vertex i.

We first compute a multimodal similarity matrix $\mathbf{M} \equiv [m_{ij}]_{n \times n}$

$$m_{ij} = \begin{cases} s_{ij} & \text{if } j \in \mathcal{N}_k(i) \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Each row i of $\mathbf{M}$ contains exactly k entries that represent weighted links to the k-nearest neighbors $\mathcal{N}_k(i)$ of image i. $\mathbf{P}$ is then derived by normalization:

$$p_{ij} = \frac{m_{ij}}{\sum_j m_{ij}} \tag{4}$$

Now let $[\mathbf{x}_t(i)]_{n \times 1}$ denote the so called state distribution containing the all probabilities of arriving at image (or vertex) i at time instance t during the random walk. Those probabilities can be computed for all future time instances by iteratively applying the transition matrix $\mathbf{P}$:

$$\mathbf{x}_{t+1}(j) = \alpha \sum_{i=1}^{n} \mathbf{x}_t(i) p_{ij} + (1 - \alpha)\mathbf{v}(j), \tag{5}$$

where $\mathbf{v}$ denotes a $n \times 1$ *bias vector* and $\alpha \in [0, 1]$ is a linear fusion variable.

While the bias vector is in most cases used to ensure irreducibility[1] and therefore convergence of the random walk, it also allows to assign some nodes a higher importance prior to the actual ranking procedure. Thus we examine two different bias settings in our experiments (see Section 4.3):

- A uniform bias where we assign the same bias value to all nodes.
- A non-uniform bias where the bias values correspond to a simple initial estimate of the importance of each image.

If no auxiliary knowledge is available the bias vector $\mathbf{v}$ may be initialized uniformly. Typically all entries are set to $1/n$ with n denoting the total number of images in the database. This ensures irreducibility and therefore convergence of the random walk process. On the other hand an initial estimate may improve the overall results. For instance the visual similarity computed from the image content itself could be used

[1] A directed graph is called irreducible, if there exists a path from each vertex to all others.

to obtain an initial relevance estimate for the images. However, as our experiments show (see Section 4.3), exploiting the tag lists alone provides more accurate retrieval results than using the visual similarity to compare images. Therefore we calculate a simple initial estimate based on text features derived from the images' tags and a *kernel density estimator (KDE)* [20]. In our implementation we use this initial estimate only to set the bias values for the L images having the largest values. We will determine a good choice for the parameter L in our experiments (see Section 4).

We will now show how the stationary distribution of the random walk process can be computed. Therefore we first re-write Equation 5 more compactly:

$$\begin{aligned}\mathbf{x}_{t+1}^T &= \alpha \mathbf{x}_t^T \mathbf{P} + (1-\alpha)\overbrace{(\mathbf{x}_t^T \mathbf{e})}^{1}\mathbf{v}^T \\ &= \mathbf{x}_t^T(\alpha \mathbf{P} + (1-\alpha)\mathbf{e}\mathbf{v}^T) \\ &= \mathbf{x}_t^T \overline{\mathbf{P}} \end{aligned} \tag{6}$$

where $\mathbf{e}$ is a $n \times 1$ vector with all entries set to 1. Note that, $\mathbf{v}^T = (\mathbf{x}_t^T \mathbf{e}\mathbf{v}^T)$ as $\sum_i \mathbf{x}_t(i) = 1$.

Thus, according to Equation 6, we can express the random walk as matrix-vector multiplication, also known as the *power method*. With t going to infinity and assuming irreducibility of $\overline{\mathbf{P}}$, the random walk converges to a (unique) stationary state distribution $\mathbf{x}_\pi$:

$$\mathbf{x}_\pi^T = \mathbf{x}_\pi^T \overline{\mathbf{P}} \Longleftrightarrow \mathbf{x}_\pi = \overline{\mathbf{P}}^T \mathbf{x}_\pi \tag{7}$$

That is, the distribution $\mathbf{x}_\pi$ does not change anymore during subsequent iterations. $\mathbf{x}_\pi$ is then equivalent to the dominant eigenvector of matrix $\overline{\mathbf{P}}^T$ [9].

2.3 Query-Independent Image Retrieval

We have not stated yet which images we use to build our multimodal graph. As we consider a query-by-term ranking task, two different cases are possible. Both will be experimentally evaluated in Section 4.3.

First we test a *query-dependent* approach where we compute the relevance scores by a random walk only for a subset of images. Typically one would pre-filter the images in the database to keep only those likely showing the query content. Thus we limit the set of candidate images used for the graph construction by keeping only those being labeled with the query term. This procedure reduces the size of the resulting graph. It may also lead to a more reliable link structure of the graph as image similarities may be estimated more accurately for this subset than for all images.

However, in the query-dependent case a graph needs to be built separately for every possible query. Therefore we explore a second, *query-independent* ranking approach. Here the relevance of an image is computed only once independent of the query term. No pre-filtering of the images is necessary and similarities are computed between all images regardless of their annotation. Thus we may experience a deterioration of the retrieval results. On the other hand, since this approach does not depend on a certain query tag, all computations can be performed off-line in advance. To return the result for a certain query term w, we then simply need to look up the relevance score of those images of the database that are within the subset labeled with w.

3. IMPLEMENTATION

3.1 Image Description

As stated above, we build our image graph using an image similarity measure based on two different modalities: The visual similarity of the image content and the textual similarity of their associated tags. To be able to apply Equation 1, we need a fixed size vector representation of each image for both modalities. In this work we chose an image description for both the visual and textual image description based on the *probabilistic Latent Semantic Analysis (pLSA)*. The pLSA enables learning an abstract high-level description from the occurrence counts of low-level features like words (tags) or quantized basic visual features (commonly referred to as visual words). We derive a representation that is low-dimensional and thus, once computed, very efficient. Moreover, compared to directly using low-level image features, it describes the images' content with fewer noise by overcoming some issues, for instance polysemy and synonymy in the text features. In the visual case, Lienhart and Slaney [14] showed that comparing topic vectors yields better results than directly comparing bag-of-word histograms.

The pLSA was originally introduced by Hofmann [6] in the context of text document modeling and retrieval. The key concept of the pLSA model is to map the high-dimensional word distribution or word count vector of a document to a lower dimensional *topic distribution* (also called *aspect vector*). To achieve this, pLSA introduces a latent, i.e. unobservable, topic layer between the documents and the words. It is assumed that each document (in our case an image) consists of a mixture of multiple topics and that the occurrences of words (i.e., visual words in images or tags associated with images) stem from the topics in the respective mixture. This generative model is expressed by the following probabilistic model:

$$p(d_i, w_j) = p(d_i) \sum_K p(z_k|d_i) p(w_j|z_k) \tag{8}$$

where $p(d_i)$ denotes the probability of a document d_i of the collection to be picked, $p(z_k|d_i)$ the probability of a topic z_k given the current document, and $p(w_j|z_k)$ the probability of a visual word w_j given a topic. K denotes the number of topics.

Although the latent topics describe the content of images, only the occurrence of words in tag lists or visual words in images can be observed in practice. To learn the pLSA model, i.e. its distributions $p(z_k|d_i)$ and $p(w_j|z_k)$, the Expectation-Maximization algorithm [3, 6] is applied. Note that the learning procedure is completely unsupervised and therefore the topics themselves are defined automatically.

Once a topic mixture $p(z_k|d_i)$ is derived for each document d_i, a high-level representation based on the respective modality has been found. The entries of the topic vector denote to which extend an image depicts a certain topic. As we commonly choose the number of concepts in our model to be much smaller than the number of distinctive words this representation is low-dimensional. The K-dimensional topic mixture vector is then used to compute image similarities as in Equation 1 and 2.

3.1.1 Tag Features

In order to learn the pLSA model on the image tags we need to define a finite vocabulary. We consider only the most commonly used words, i.e. our vocabulary consists of all tags in our dataset that are used by more than 100 users. Further we do not allow tags with numbers. The resulting vocabulary consisting of 2977 distinct words was applied to derive a bag-of-words description for each image based on its associated tags. These word count vectors are then used to compute a pLSA model with 200 topics. Thus we derive a 200-dimensional tag-based image description for our text modality. We empirically chose 200 topics as a trade-off between a low-dimensional vector and a more detailed representation.

Note that the resulting topic vector $P(z_k|d_i)$ computed by the pLSA is a compact representation that allows to compute similarities by simple vector operations in contrast to tag lists. Additionally the topic distribution also allows to match synonyms and homonyms when comparing images.

3.1.2 Visual Features

To apply the pLSA model in the visual domain we consider each image as a single visual document. The pLSA can be applied directly to image tags, as image tags consist of words. However, for our visual features we need comparable elementary parts called visual words. These visual words are computed by quantizing local feature descriptors extracted from image regions.

We determine the image regions by applying dense sampling with a vertical and horizontal step size of 10 pixels across an image pyramid created with a scale factor of 1.2. SIFT descriptors [16] computed over a region of 41×41 pixels are then used to describe the local image content in a scale and orientation invariant fashion. It should be noted that any other feature detector or descriptor could be used instead. Prior to the actual feature extraction, images are scaled down to a maximum side length of 500 pixel.

Quantization is performed with a vocabulary tree [17] in order to support large vocabulary sizes. The vocabulary tree is computed by repeated k-means clustering that hierarchically partitions the feature space. Such a hierarchical approach overcomes two major problems of the traditional direct k-means clustering in cases where k is large. Firstly, during vocabulary learning, applying the k-means algorithm repeatedly with small k is computationally more efficient than running it only once with larger k. Secondly, the mapping of visual features to discrete words is very fast. In our experiments we constructed a visual vocabulary consisting of 10,000 visual words.

Once the visual vocabulary is determined we map each feature vector of each image to its closest visual word. Then, we derive a bag-of-visual words representation by counting the occurrences of each visual word in the respective image. Note that this image content description does not preserve any geometric relationship between the occurrences of the visual words.

The word count vectors are then used to compute the pLSA model. Similar to the pLSA on tags we used 200 topics leading to a 200-dimensional description of each image's content based on visual features.

3.2 Nearest Neighbor Search

As the random walk requires to find the k-nearest neighbors of *each* image, a naive implementation would result in computing similarities between all image pairs and sorting those similarities for each image. This would result in at least $O(n^2)$ comparisons which certainly limits the scalability of our approach, especially in the query-independent case (see Section 2.3).

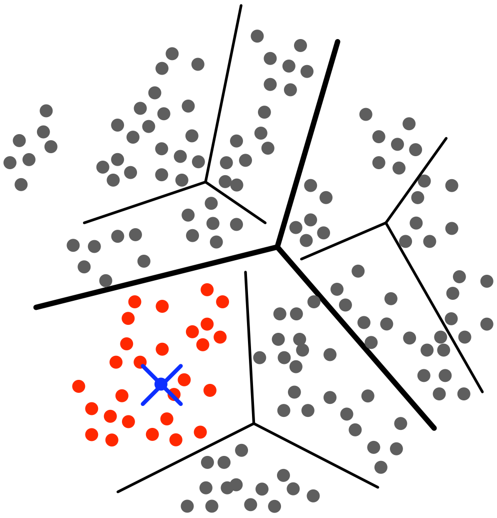

Figure 3: The hierarchically partitioned topic space. For a given query vector (blue dot) the nearest cluster is found and then the nearest neighbors are computed by comparing the vectors belonging to the same cluster (red dots).

Therefore we propose an approach that hierarchically partitions the topic space. The nearest neighbor search of an image is then limited to such a subspace, i.e. to a subset of images in our database. This way we reduce the number of image comparisons required for the graph construction to a linear amount depending on the cluster sizes.

To determine such a partition we cluster the images' representations, in our case their topic vectors, hierarchically by applying the k-means algorithm recursively. Each of the resulting cascaded subspaces is represented by its respective cluster centroid, i.e. the mean vector over all representations in the cluster. The resulting tree allows searching the topic space efficiently for the nearest cluster for a given image description. This is done by propagating down the tree, i.e. comparing the descriptor vector to the centroids at each level and choosing the closest. Once the nearest cluster is determined the image only needs to be compared to all members of that cluster (see Figure 3).

The more clusters emerge from the hierarchical clustering the faster the k-nearest neighbor search will be as clusters tend to be smaller and thus fewer vectors need to be compared. On the other hand, a larger number of subsets may lead to a performance degradation due to the introduction of inaccuracies at subspace borders. For our dataset consisting of roughly 260,000 images (see Section 4.1) we choose an hierarchical clustering into two clusters on each level, thus we construct a binary tree. As long as a cluster consists of more than 25,000 associated vectors it is further recursively sub-divided. This procedure results into 15 subspaces/image subsets with cluster sizes ranging from 10,740 to 23,889 vectors. We use the multimodal similarity measure as defined in Equation 2 as our distance measure for computing the clusters.

Figure 4: Some randomly picked images from the image set consisting of 28 categories.

As this k-nearest neighbor search is an approximation, we also empirically measured the intersection of the nearest neighbors computed with and without approximation in order to get an idea of the scale of the inaccuracies we have introduced by the approximation. Surprisingly only 34% identical images could be found in our experiments. However, this has no negative impact on the overall performance of our query-independent system as shown in Section 4.3.

4. EVALUATION

4.1 Database

We evaluate our presented approach experimentally on a dataset consisting of 261,901 Flickr images. To derive this dataset we downloaded up to 10,000 images from each of the following 28 categories: *aircraft, beach, bicycle(s), bird(s), boat, bottle(s), building, bus(es), car(s), cat(s), chair(s), city, coast, cow(s), desert, dog, forest, horse(s), motorcycle(s), mountain(s), people, potted plant, sheep, sofa, street(s), table(s), tree(s), tv.* As we downloaded images according to their user tags, images with multiple tags can be part of multiple categories. Note that the dataset contains objects as well as scene categories. Some example images from the dataset can be seen in Figure 4.

The database has not been cleaned or post-processed. A manual statistical analysis has shown that on average about 40% of the images are mislabeled with tags that do not describe the depicted image content (see Section 4.3).

We have also stored the metadata associated with the images including their tags. We counted 136,371 unique tags in our collection, leading to a vocabulary of size 2977 after filtering (see Section 3.1).

4.2 Methodology

As no ground truth is available for this large-scale image database, test users are required to rate the performance of our system. Therefore we evaluate our ranking system and the different parameter settings by conducting several user studies. For each category of our image database the most frequently occurring tag is selected as the query term resulting into 28 query terms. Then we ask 10 participants to judge the top 19 results as ”relevant”, ”somewhat relevant” or ”not relevant” to the query.

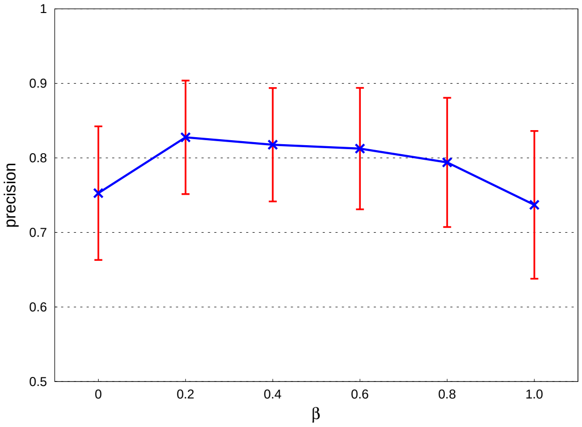

Figure 5: The results shown for different weights β of the visual and textual modalities.

We use the following scoring to get a quantitative performance measure: An image considered being relevant receives 1 point, an image considered as somewhat relevant receives 0.5 points. All other images get 0 points. A mean score is calculated for each user; the mean over all users’ means yields the final score of the parameter configuration/system being evaluated.

4.3 Experiments

In our first experiments we examine the influence of several parameters in our image ranking approach using the query-dependent setting. Then we compare the best query-dependent image ranking method to two baselines. Finally we compare the result of the query-dependent image ranking method to the proposed query-independent algorithm.

First, we determine the optimal weight β for the two modalities in our multimodal similarity measure (see Section 2.1). Therefore we vary β and keep all other parameters fixed: We chose $k = 250$ nearest neighbors and a non-uniform bias to weight the top 500 positive entries for building the graph. Figure 5 shows the results of our user studies. As can be seen, the best results are obtained for $\beta = 0.2$, i.e. the text modality receives a four times larger weight than the visual modality. Comparing images by their associated tags seems to be slightly more reliable than comparing them by visual content. However, fusing both similarities as proposed improves the results over using only one single modality to measure similarity. Hence, we fix β to 0.2 for the following experiments.

In our second experiment we vary the number of nearest neighbors k used to establish the link structure in our image graph. Since we assume larger values of k to introduce unreliable links to the graph, we choose $k \in \{100, 250, 500\}$. When changing k we need to adapt the bias in order to weight the nodes with the same magnitude. To avoid this issue, we chose a uniform bias to obtain comparable results. As can be seen from Figure 6 the best results have been obtained by using $k = 100$ and $k = 250$ neighbors. Using more neighbors may introduce noise that degrades the quality of the link structure.

In our third experiment we evaluate the impact of the bias. Therefore we compare a uniform bias to non-uniform biases as described in Section 2.2. For the non-uniform bias we

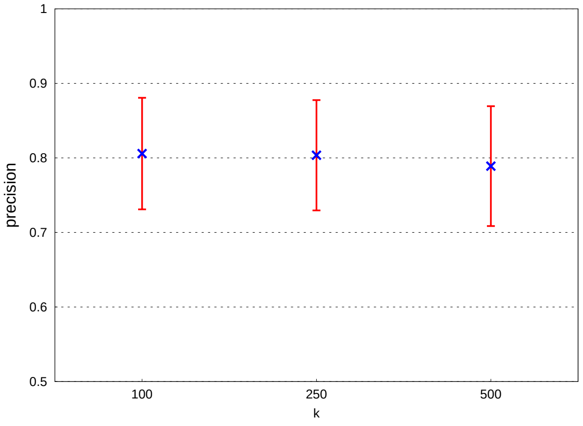

Figure 6: The results shown for a varying number of k-nearest neighbors of each node.

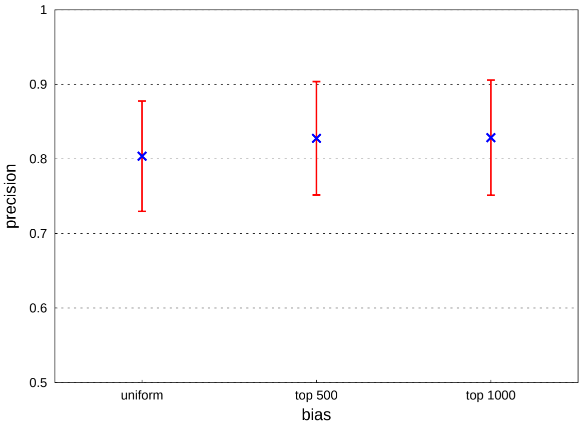

Figure 7: The results for different types of biases.

examine two different values for the parameter L, $L = 500$ and $L = 1000$, indicating the number of images biased with their respective initial estimates. The other parameters, $k = 250$ and $\beta = 0.2$, have been chosen according to the results of the previous two experiments. As illustrated in Figure 7 we obtain the highest score for a non-uniform bias where either the top 500 or top 1000 most important images have been biased. However, the influence of the bias is small, especially since we have empirically chosen $\alpha = 0.9$ (see Section 2.2).

Having determined the optimal parameter setting, we now compare our query-dependent approach to two different baselines. As a worst-case baseline we draw random images from each of the 28 tag categories. This baseline reflects the average relevance of images within the tag categories included in our user studies. We also compare our system to a state-of-the-art approach described in [15]. Here the tags associated with an image are first re-ranked using both, image content and associated tags. Given a query tag w, the images are then ranked according to a relevance score $r(x)$ purely relying on the re-ranked tag list:

$$r(x) = -\tau(w) + 1/|x|, \quad (9)$$

where $\tau(w)$ denotes the ranked position of w in the tag list of length $|x|$ associated with image x.

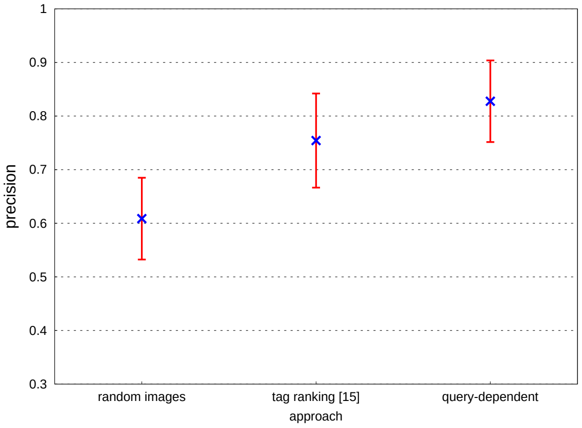

Figure 8: A performance comparison of different image ranking approaches.

The results of our user study are depicted in Figure 8. In Figure 9 we also show the scores separately for each query term. One can see that the proposed multimodal query-dependent image ranking outperforms both baselines in average and in most categories. Note that, while the tag ranking approach performs well, we consider it hardly usable for practical use, as it has a strong focus on very short tag lists.

In our last experiment we evaluate the proposed query-independent approach. Due to the much larger number of images within the image graph in the query-independent setting, we empirically set the number of neighbors k to 500 to build the graph. As described in Section 3.2 we performed the k-nearest neighbor search by utilizing the hierarchically clustered topic space. The results show that the query-independent approach allows to compute the relevant images in a scalable way with only a minor loss of precision. Moreover, as can be seen in Figure 10, the query-independent approach unexpectedly performs better when the tree is used to approximate the k-nearest neighbors than determining the neighbors by direct computation. This might be due to the clustering as it divides the vector space of our image representation largely based on the tags of our images. Therefore irrelevant neighbors might be excluded as in the query-dependent case before the absolute distance between images is taken into account.

For both the query-dependent and query-independent approach we are able to retrieve relevant and at the same time diverse result images for different query terms such as objects, sceneries and animals. Figures 11 and 12 show the top 16 retrieved images for the query terms *building* and *coast*. We used the query-dependent approach with $k = 250$ neighbors, $\beta = 0.2$ and a non-uniform bias setting the top 500 entries. For the query-independent system, Figure 13 shows the results of using *sheep* as query term, with $k = 500$ neighbors, $\beta = 0.2$ and a uniform bias.

5. CONCLUSION

We have presented an image ranking method based on random walk on an image graph. This graph is built from images and their similarities among each other. We proposed to use a multimodal similarity measure to find nearest neighbors of images. Our experiments show that combining

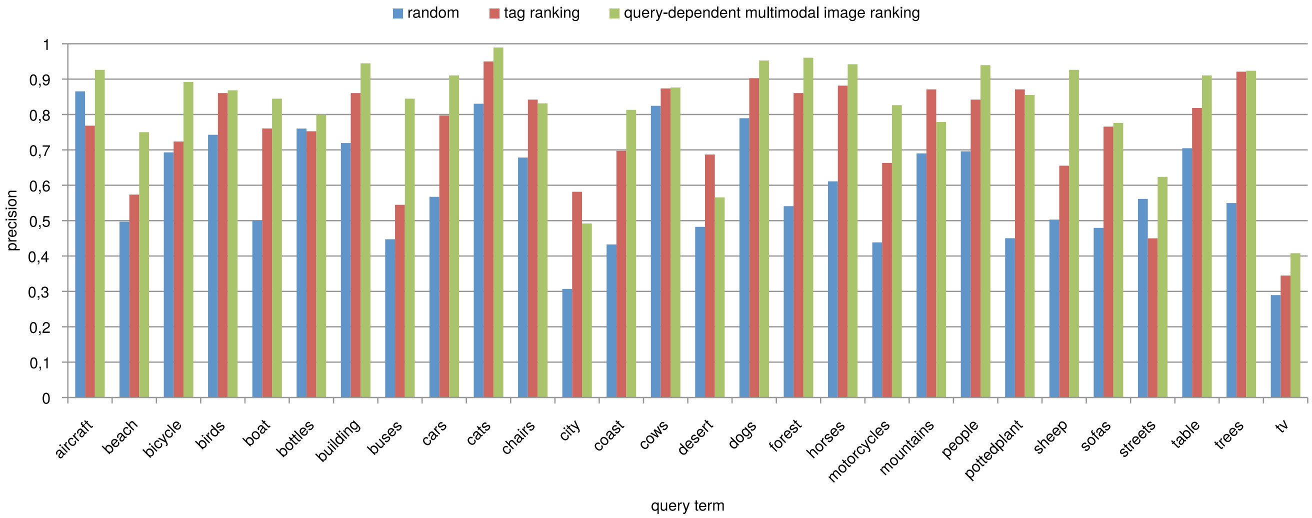

Figure 9: Comparison between our query-dependent approach and both baselines per query term.

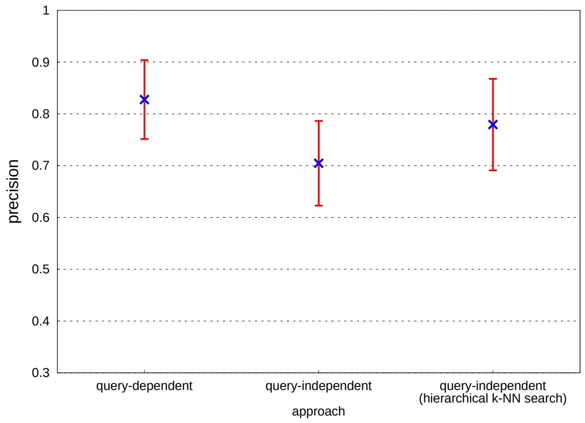

Figure 10: The query-dependent approach compared to the query-independent approach.

more than one modality improves the system performance significantly. Moreover we show a query-independent ranking approach that allowed to compute a global ranking score prior to a certain query. In this work we further addressed scalability issues by using a k-nearest neighbor approximation when building the graph and computing the similarities between images. Future work will evaluate the query-independent approach on larger databases, explore its scalability in more detail and evaluate the proposed approach in the context of abstract query terms.

6. REFERENCES

[1] T. L. Berg and A. C. Berg. Finding iconic images. In *The 2nd Internet Vision Workshop at IEEE CVPR*, 2009.

[2] D. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the world's photos. In *Proc. 18th International World Wide Web Conference*, 2009.

[3] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[4] D. Grangier and S. Bengio. A discriminative kernel-based approach to rank images from text queries. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(8):1371–1384, Aug. 2008.

[5] X. He, W.-Y. Ma, and H. Zhang. Imagerank: spectral techniques for structural analysis of image database. In *ICME '03: Proceedings of the 2003 International Conference on Multimedia and Expo*, pages 25–28, Washington, DC, USA, 2003. IEEE Computer Society.

[6] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42, Numbers 1-2:177–196, 2001.

[7] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking through random walk over

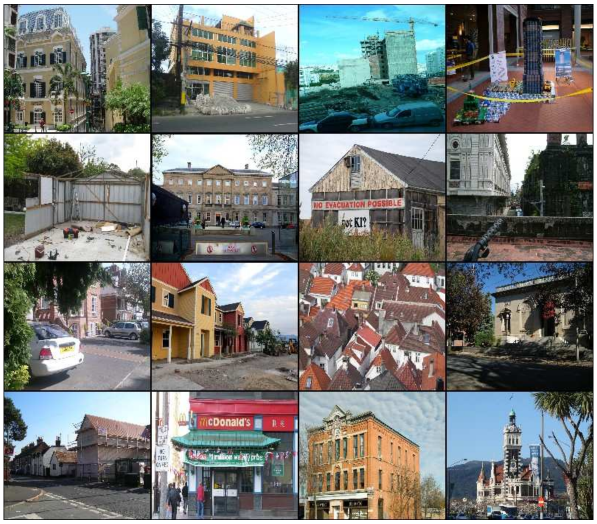

Figure 11: Top 16 (query-dependent) retrieval results for query term *building*.

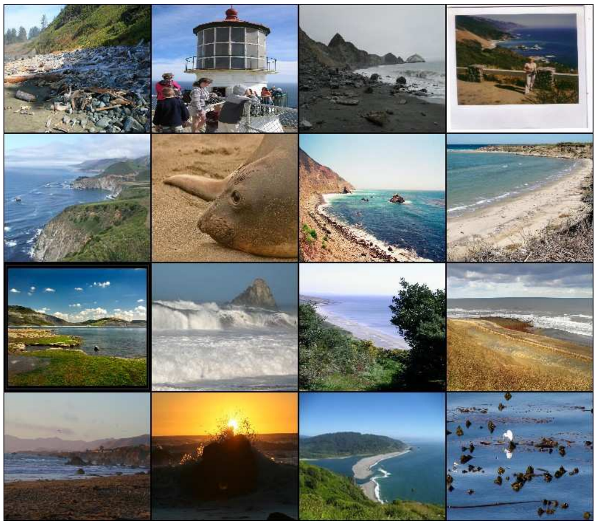

Figure 12: Top 16 (query-dependent) retrieval results for query term *coast*.

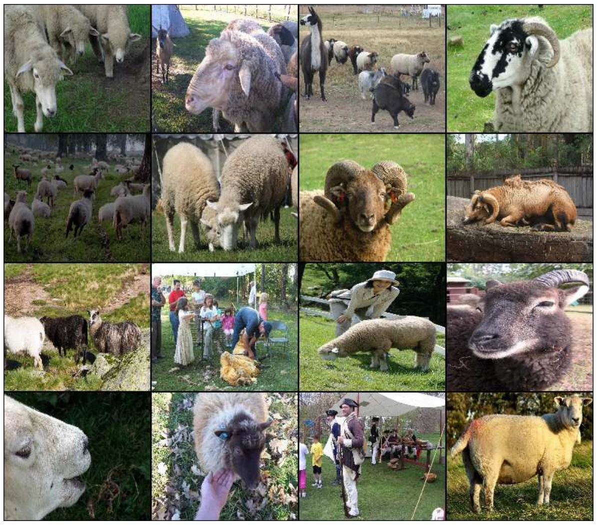

Figure 13: Top 16 (query-independent) retrieval results for query term *sheep*.

document-level context graph. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 971–980, New York, NY, USA, 2007. ACM.

[8] Y. Jing and S. Baluja. Visualrank: Applying pagerank to large-scale image search. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(11):1877–1890, 2008.

[9] S. Kamvar, T. Haveliwala, C. Manning, and G. Golub. Extrapolation methods for accelerating pagerank computations. In *In Proceedings of the Twelfth International World Wide Web Conference*, pages 261–270. ACM Press, 2003.

[10] L. S. Kennedy and M. Naaman. Generating diverse and representative image search results for landmarks. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 297–306, New York, NY, USA, 2008. ACM.

[11] X. Li, C. G. Snoek, and M. Worring. Learning tag relevance by neighbor voting for social image retrieval. In *MIR '08: Proceeding of the 1st ACM international conference on Multimedia information retrieval*, pages 180–187, New York, NY, USA, 2008. ACM.

[12] X. Li, C. Wu, C. Zach, S. Lazebnik, and J.-M. Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In D. A. Forsyth, P. H. S. Torr, and A. Zisserman, editors, *ECCV (1)*, volume 5302 of *Lecture Notes in Computer Science*, pages 427–440. Springer, 2008.

[13] R. Lienhart, S. Romberg, and E. Hörster. Multilayer plsa for multimodal image retrieval. In *ACM International Conference on Image and Video Retrieval (CIVR)*, 2009.

[14] R. Lienhart and M. Slaney. pLSA on large scale image databases. *IEEE International Conference on Acoustics, Speech and Signal Processing 2007 (ICASSP 2007)*, Vol. IV:1217–1220, 2007.

[15] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang. Tag ranking. In *18th International World Wide Web Conference*, pages 351–351, April 2009.

[16] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60 (2):91–110, 2004.

[17] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:2161–2168, 2006.

[18] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[19] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Gcap: Graph-based automatic image captioning. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04. Conference on*, pages 146–146, June 2004.

[20] E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[21] R. Raguram and S. Lazebnik. Computing iconic summaries of general visual concepts. *Computer Vision and Pattern Recognition Workshop*, 0:1–8, 2008.

[22] S. Romberg, E. Hörster, and R. Lienhart. Multimodal plsa on visual features and tags. In *IEEE International Conference on Multimedia and Expo (ICME)*, 2009.

[23] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting image databases from the web. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, Oct. 2007.

[24] G. Wang and D. Forsyth. Object image retrieval by exploiting online knowledge resources. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[25] Y.-T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T.-S. Chua, and H. Neven. Tour the world: building a web-scale landmark recognition engine. In *Proc. of ICCV*, Miami, Florida, U.S.A, June, 2009.