Hybrid Approach for Image Search Reranking

Sabitha M G[1], R Hariharan[2]

[1]PG Scholar, Department of CSE, Maharaja Prithvi Engineering College
Avinashi, Tamilnadu, India

[2]Asst.Professor, Department of CSE, Maharaja Prithvi Engineering College
Avinashi, Tamilnadu, India

Abstract*: Web image retrieval is a challenging task that requires efforts from image processing, link structure analysis, and web text retrieval. In this paper, we propose a re-ranking method to improve web image retrieval by reordering the images retrieved from an image search engine. The re-ranking process is based on a relevance model, which is a probabilistic model that evaluates the relevance of the HTML document linking to the image, and assigns a probability of relevance. The top-ranked images are used as (noisy) training data and an SVM visual classifier is learned to improve the ranking further. We investigate the sensitivity of the cross-validation procedure to this noisy training data. The principal novelty of the overall method is in combining text/metadata and visual features in order to achieve a completely automatic ranking of the images. Human supervision is introduced to learn the model weights offline, prior to the online reranking process The experiment results showed that the re-ranked image retrieval achieved better performance than original web image retrieval, suggesting the effectiveness of the re-ranking method. The relevance model is learned from the Internet without preparing any training data and independent of the underlying algorithm of the image search engines. The re-ranking process should be applicable to any image search engines with little effort..*

Keywords: Support vector machine, Visual Information Retrieval, radial basis function

1. Introduction

The objective of this work is to retrieve a large number of images for a specified object class from the browser. A multimodal approach employing text, metadata, and visual features is used to gather many high-quality images from the Web. Candidate images are obtained by a text-based search querying on the object identifier. We compare three different approaches to downloading images from the Web. The first approach, named Web Search, second image search starts from Google image search (rather than Web search). The third approach, Google Images, includes only the images directly returned by Google image search (a subset of those returned by Image Search).

The task is then to remove irrelevant images and re-rank the remainder. First, the images are re-ranked based on the text surrounding the image and metadata features. A number of methods are compared for this re-ranking. Second, the top-ranked images are used as (noisy) training data and an SVM visual classifier is learned to improve the ranking further. We investigate the sensitivity of the cross-validation procedure to this noisy training data Based on the images in the initial result, visual prototypes are generated that visually represent the query.

The existing web image search engines, including Bing [1], Google [2], and Yahoo! [3], retrieve and rank images mostly based on the textual information associated with the image in the hosting web pages, such as the title and the sur-rounding text. While text-based image ranking is often effective to search for relevant images, the precision of the search result is largely limited by the mismatch between the true relevance of an image and its relevance inferred from the associated textual descriptions [4].

Current approaches to object category recognition require datasets of training images to be manually prepared, with varying degrees of supervision. This present an approach that can learn an object category from just its name, by utilizing the raw output of image search engines available on the Internet. It develops a new model, TSI-pLSA, which extends pLSA (as applied to visual words) to include spatial information in a translation and scale invariant manner. This approach can handle the high intra-class variability and large proportion of unrelated images returned by search engines. Evaluate the models on standard test sets, showing performance competitive with existing methods trained on hand prepared datasets.

The recognition of object categories is a challenging problem within computer vision. The current paradigm consists of manually collecting a large training set of good exemplars of the desired object category; training a classifier on them and then evaluating it on novel images, possibly of a more challenging nature. It propose a different perspective on the problem. There is a plentiful supply of images available at the typing of a single word using Internet image search engines such as Google, and propose to learn visual models directly from this source.It provide an approach that can learn an object category from just its name, by utilizing the raw output of image search engines available on the Internet This approach can handle the high intra-class variability and large proportion of unrelated images returned by search engines. This has proposed the idea of training using just the objects name by bootstrapping with an image search engine. The training sets are extremely noisy yet, for the most part, the results are competitive (or close to) existing methods requiring hand gathered collections of images.

The paper is organized as follows. In Section II, we briefly review the related work on visual reranking. In Section III, we provide an overview of our proposed method. The experimental results are presented and analyzed in Section V, while Section VI concludes the paper with a brief overview of the main results of the paper and the prospects for future work.

2. Related work

The methods for image search reranking can be classified into supervised and unsupervised ones, according to whether human labeled data has been used to derive the reranking model or not. The unsupervised reranking methods do not rely on human labeling of relevant images but require prior assumptions on how to employ the information contained in the underlying text- based result for reranking.

3. Hybrid Image Reranking

3.1 System Framework

When an image search in search engines, that corresponding images are loaded in that time, meanwhile among them there is a uncategorized images are also spotted. However, producing such databases containing a large number of images and with high precision is still an arduous manual task. Image clusters for each topic are formed by selecting images where nearby text is top ranked by the topic. A user then partitions the clusters into positive and negative for the class. Second, images and the associated text from these clusters are used as exemplars to train a classifier based on voting on visual (shape, color, and texture) and text features.

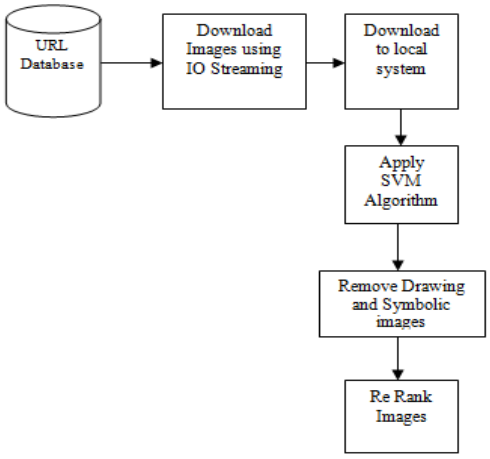

Figure 1: System Architecture of Image Reranking framework

We compare three different approaches to downloading images from the Web.

The first approach, named Web Search, submits the query word to Google Web search and all images that are linked within the returned Web pages are downloaded. Google limits the number of returned Web pages to 1,000, but many of the Web pages contain multiple images, so in this manner, thousands of images are obtained.

The second approach, Image Search, starts from Google image search (rather than Web search). Google image search limits the number of returned images to 1,000, but here, each of the returned images is treated as a "seed"—further images are downloaded from the Webpage where the seed image originated.

The third approach, Google Images includes only the images directly returned by Google image search (a subset of those returned by Image Search). The query can consist of a single word or more specific descriptions such as "penguin animal" or "penguin OR penguins." Images smaller than 120 _ 120 are discarded. In addition to the images, text surrounding the image HTML tag is downloaded, together with other metadata such as the image filename.

Image Search gives a very low precision (only about 4 percent) and is not used for the harvesting experiments. This low precision is probably due to the fact that Google selects many images from Web gallery pages which contain images of all sorts. Google is able to select the in-class images from those pages, e.g., the ones with the object-class in the filename; however, if we use those Web pages as seeds, the overall precision greatly decreases. Therefore, we only use Web Search and Google Images, which are merged into one data set per object class. Table 2 lists the 18 categories downloaded and the corresponding statistics for in-class and non-class images. The overall precision of the images downloaded for all 18 classes is about 29 percent. Now describe the re-ranking of the returned images based on text and metadata alone. Here, we follow and extend the method proposed by using a set of textual attributes whose presence is a strong indication of the image content.

The goal is to re-rank the retrieved images. Each feature is treated as binary: "True" ifit contains the query word (e.g., penguin) and "False" otherwise. To re-rank images for one particular class (e.g., penguin), we do not employ the whole images for that class. Instead, we train the classifier using all available annotations except the class we want to re-rank. This way, we evaluate performance as a completely automatic class independent image ranker, i.e., for any new and unknown class, the images can be re-ranked without ever using labeled ground-truth knowledge **(**images are divided into three categories: 1.Good, 2.Ok, 3.non-class**)** of that class.

It is interesting to note that the performance is comparable to the case of filtered images. This means that the learned visual model is strong enough to remove the drawings and symbolic images during the ranking process. Thus, the filtering is only necessary to train the visual classifier and is not required to rank new images, However, using unfiltered images during training decreases the performance significantly, where training with filtered images is a lot worse than with unfiltered images.

3.2 Implementation and Result

Data collection

We compare three different approaches to downloading images from the Web. The first approach, named Web Search, submits the query word to Google Web search and all images that are linked within the returned Web pages are downloaded. Google limits the number of returned Web pages to 1,000, but many of the Web pages contain multiple images, so in this manner, thousands of images are obtained. The second approach, Image Search, starts from Google image search (rather than Web search).

Google image search limits the number of returned images to 1,000, but here, each of the returned images is treated as a seed"—further images are downloaded from the Webpage

where the seed image originated. The third approach, Google Images, includes only the images directly returned by Google image search (a subset of those returned by Image Search). The query can consist of a single word or more specific descriptions such as “penguin animal” or “penguin OR penguins.” Images smaller than 120_120 are discarded. In addition to the images, text surrounding the image HTML tag is downloaded, together with other metadata such as the image filename.

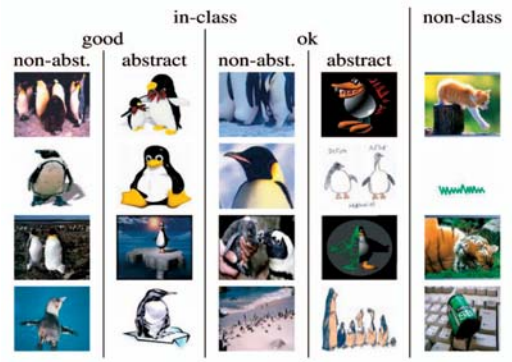

Figure 2: Ground-truth annotation

In a similar manner, images are divided into three categories: **in-class-good.** Images that contain one or many class instances in a clearly visible way (without major occlusion, lighting deterioration, or background clutter, and of sufficient size).**in-class-ok.** Images that show parts of a class instance, or obfuscated views of the object due to lighting, clutter, occlusion, and the like.**nonclas** Images not belonging to in-class.

The good and ok sets are further divided into two subclasses: **abstract.** Images that do not resemble realistic natural objects (e.g., drawings, nonrealistic paintings, comics, casts, or statues).**nonabstract.** Images not belonging to the previous class.

Removing Drawings and Symbolic Images

Since we are mostly interested in building databases for natural image recognition, we ideally would like to remove all abstract images from the downloaded images. However, separating abstract images from all others automatically is very challenging for classifiers based on visual features. Instead, we tackle the easier visual task of removing drawings and symbolic images. These include: comics, graphs, plots, maps, charts, drawings, and sketches, where the images can be fairly simply characterized by their visual features. Their removal significantly reduces the number of non-class images, improving the resulting precision of the object class data sets (overall precision goes from 29 to 35 percent). Filtering out such images also has the aim of removing this type of abstract image from the in-class images.

Learning the filter. We train a radial basis function Support Vector Machine (SVM) on a hand-labeled data set. After the initial training, no further user interaction is required. In order to obtain this data set, images were downloaded using Image Search with one level of recursion (i.e., Web pages linked from “seed” Web pages are also used) with queries such as “sketch” or “drawing” or “draft.” The goal was to retrieve many images and then select suitable training images manually.

Visual Information Retrieval

Visual Information Retrieval (VIR) is a relatively new field of research in Computer Science and Engineering. As in conventional information retrieval, the purpose of a VIR system is to retrieve all the images (or image sequences) that are relevant to a user query while retrieving as few non-relevant images as possible. The emphasis is on the retrieval of information as opposed to the retrieval of data. Similarly to its text-based counterpart a visual information retrieval system must be able to interpret the contents of the documents (images) in a collection and rank them according to a degree of relevance to the user query. The interpretation process involves extracting (semantic) information from the documents (images) and using this information to match the user needs.

Progress in visual information retrieval has been fostered by many research fields, particularly: (text-based) information retrieval, image processing and computer vision, pattern recognition, multimedia database organization, multidimensional indexing, psychological modeling of user behavior, man-machine interaction, among many others.

Feature Extraction Models

This section describes the color models used in the experiments and explains how the color information of the partition- and region-based approaches can be extracted from an image.

Color models

Some color models, such as HSV and CIE L*u*v*, are proposed to overcome this problem. Their color characteristics are separated into three parts: hue, lightness, and saturation, which make them more consistent with human vision. In our approach, we choose the HSV color model to represent the color information of an image. In the HSV color model, the color characteristics are separated into three parts: hue, saturation, and value. Because the total number of colors in the HSV color model is too high, it is necessary to partition the whole HSV color space into several sub-spaces where similar colors are associated together. The color values of the original pixels in an image are represented by the R, G, and B values, so that a transformation from the RGB to the HSV color model is necessary. It can be accomplished by the algorithm proposed. The RGB color model is widely used to represent digital images on most computer systems. However, the RGB color model has a major drawback on the similarity measure.

Color information in the partition-based approach

To extract the color information of the images in the partition-based approach, we first divide an image into m×n equal-sized blocks. Next, a dominant color is extracted from each block, which is the color that has a maximum number of pixels in a block. A two-stage examination is used to decide whether it can be a representative color. In the first stage, only the number of pixels of the dominant color is

checked. If it is larger than a predefined threshold, the color is designated as the representative color of a block and the second-stage examination need not be performed. Otherwise, we add the number of pixels of the neighbor colors of the dominant color and check the threshold again. The neighbor colors are the colors adjacent to the dominant color in the color space

Color information in the region-based approach

The representative colors of the blocks in an image are used to extract its color information in the region-based approach. A region is obtained by grouping adjacent blocks having the same representative color or neighbor colors. Therefore a region may have more than one representative color. The color associated with the most blocks in a region is selected as its representative color. The adjacent blocks, which have the same representative color, are grouped into a region first. Next, if the representative colors of two adjacent regions are neighbor colors, they are grouped into a larger region. The size of a region is the total number of blocks contained in the region.

Note that the grouping order between regions has to be fixed, because the representative color of regions may change with the grouping order. In our approach, we group regions from the top-left to the bottom-right corner. For each region we record its shape, size, and the representative color as its properties. The shape is represented by the ratio of the short edge to the long edge of the MBR (minimum bounding rectangle) of the region. Because of the restricted order of division, we can deal with rotated objects to some degree. Moreover, smaller regions (with a size less than a threshold) are dropped. The threshold depends on the image size. In our database the image size is 192*128 pixels. Therefore, we set the threshold at 2048 pixels (about 10% of the image size).

Ranking on Visual features

The text re-ranking of associates a posterior probability with each image as to whether it contains the query class or not. The problem we are now faced with is how to use this information to train a visual classifier that would improve the ranking further. The problem is one of training from noisy data: We need to decide which images to use for positive and negative training data and how to select a validation set in order to optimize the parameters of the classifier. We first describe the visual features used and then how the classifier is trained.

Visual features. We follow the approach and use a variety of region detectors with a common visual vocabulary in the bag of visual words model framework. All images are first resized to 300 pixels in width. Regions are detected using difference of Gaussians, Kadir's saliency operator, and points sampled from Canny edge points. Each image region is represented as a 72-dimensional SIFT descriptor. A separate vocabulary consisting of 100 visual words is learned for each detector using k-means, and these vocabularies are then combined into a single one of 400 words. Finally, the descriptor of each region is assigned to the vocabulary. The software for the detectors is obtained. Fuller implementation details are given and are reproduced in our implementation.

Training the Visual Classifier

At this point, we can select nþ positive training images from the top of the text-ranked list, or those that have a posterior probability above some threshold, but a subset of these positive images will be "noisy," i.e., will not be inclass. It gives an idea of the noise from the proportion of outliers. It averages 40 percent if nþ ¼ 100. However, we can assume that the nonclass images are not visually consistent—an assumption verified to some extent by the results. The case of negative images is more favorable: We select n_ images at random from all downloaded images (i.e., from all 18 classes, tens of thousands of images) and the chance of any image being of a particular class is very low.

To implement the SVM, we use the publicly available SVM light software (with the option to remove inconsistent training data enabled). Given two input images Ii and Ij and their corresponding normalized histograms of visual words and HOG, Si and Sj, this implementation uses the following _2 radial basis function (RBF) kernel, with the free kernel parameter. Thus, Cþ, and C are the three parameters that can be varied. The optimal value for these parameters is obtained by training the SVM using 10-fold cross validation. Note that we do not use the ground-truth at any stage of training, but we split the noisy training images into 10 training and validation sets.

Results for Textual/Visual Image Ranking

When evaluate different combinations of training and testing. If not stated otherwise, the text + vision system was used. For each choice, five different random selections are made for the sets used in the 10-fold cross validation, and mean and standard deviation are reported. The clear improvement brought by the visual classifier over the text-based ranking for most classes is obvious. We first investigate how the classification performance is affected by the choice of nþ and n_. It can be seen that increasing n_ tends to improve performance. It is difficult to select optimal values for nþ and n_ since these numbers are very class dependent. It indicates that using more images in the background class n_ tends to improve the performance but there is no real difference between using 150=1;000 and 250=1;000 (nþ/n_), which perform at 68:4% _ 1:9 and 68:0% _ 2:0, and thus are not significantly different. All numbers in this section report precision at 15 percent recall. It can be seen that HOG alone performs significantly worse than the bag of visual words 57:9% _ 1:8, but the combination of BOW and HOG improves the overall performance to 69:8% _ 2:1, compared to BOW alone 68:0% _ 2:0. In order to select the appropriate parameter values, we use cross validation, where the validation set is part of the nþ and n_ images together with precision at 15 percent recall as selection criterion. Combining the probabilistic outputs of text and SVM as remains an interesting addition for future work. The advantage of our system is that, once trained, we can rank images for which no metadata are available.

Basic Feature Extraction Algorithms

Many different describing features may be extracted from any two-dimensional (i.e. audio) or three-dimensional (i.e., video) signal. Depending on the objective of the

measurement, these variables may provide static or dynamic description of a stimulus. However, the application of these features towards a perceptual model depends on their relation to human models. This significance must be analyzed by comparing the results provided by raw measurements to subjective evaluations. It provides the mathematical foundation for the extraction of each of the selected features of the visual and auditory signals. Results from subjective quality-rating tests are provided for the same visual and auditory content. Finally, a detailed numerical and statistical analysis is performed that intends to determine the existence (or non-existence) of relationships between subjective and objective measurements.

Complexity and Focus of Attention Measurements

Perceptual complexity and attention workload for a particular scene may be estimated by the use of a saliency based analysis as proposed. The algorithm has been adapted for its implementation, and consists of measuring the relevant features that characterize the image and performs an independent saliency analysis on each feature map. This set of maps is combined using the appropriate weights for each feature, and a singular saliency map is determined. Three main features are extracted from the image: color, intensity and orientation.

Separate matrices are extracted for the red, green, blue and intensity channels of an RGB image. An RGB representation of color is used as it has been proposed in the saliency algorithm and also in order to avoid further processing of the input signal. In order to compensate and de-correlate intensity from color (to approach HSV representation), tuned color information is obtained from the color information by normalizing and combining the RGB channels:

The resulting feature maps are calculated by computing a set of spatial scales for each feature and then determining the necessary feature or conspicuity maps through center-surround difference operations. For intensity and color information, nine spatial scales are computed making use of Gaussian pyramids.

4. Conclusion

In this paper, Re-ranking web image retrieval can improve the performance of web image retrieval, which is supported by the experiment results. The re-ranking process based on relevance model utilizes global information from the image's HTML document to evaluate the relevance of the image. The relevance model can be learned automatically from a web text search engine without preparing any training data.

The reasonable next step is to evaluate the idea of re- ranking on more and different types of queries. At the same time, it will be infeasible to manually label thousands of images retrieved from a web image search engine. An alternative is task-oriented evaluation, like image similarity search. Given a query from Corel Image Database, can we re-rank images returned from a web image search engine and use top-rank images to find similar images in the database? We then can evaluate the performance of the re-ranking process on similarity search task as a proxy to true

References

[1] [Online]. Available: http://images.bing.com/.
[2] [Online]. Available: http://images.google.com/.
[3] [Online]. Available: http://images.yahoo.com/.
[4] Linjun Yang and Alan Hanjalic "Prototype Based image Search Reranking," in *Proc. CVPR*, 2008.
[5] L. Yang and A. Hanjalic, "Supervised reranking for web image search," in *Proc. ACM Multimedia*, 2010.
[6] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman, "Learning object categories from Google's image search," in *Proc. ICCV*, 2005, IEEE Computer Society.
[7] L.-J. Li and L. Fei-Fei, "OPTIMOL: Automatic online picture collec-Tion via incremental MOdel learning," *Int. J. Comput. Vision*, 2009.
[8] W. H. Hsu, L. S. Kennedy, and S.-F. Chang, "Video search reranking via information bottleneck principle," in *Proc. ACMMultimedia*, 2006.
[9] M. Fritz and B. Schiele, "Decomposition, discovery and detection of visual categories using topic models," in *Proc. CVPR*, 2008.
[10] R. Yan, A. G. Hauptmann, and R. Jin, "Multimedia search with pseudo- relevance feedback," in *Proc. CIVR*, 2003.
[11] W. H. Hsu, L. S. Kennedy, and S.-F. Chang, "Video search reranking through random walk over document-level context graph," in *Proc. ACM Multimedia*, 2007.
[12] Y. Jing and S. Baluja, "Visualrank: Applying pagerank to large-scale image search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 1877–1890, Nov. 2008.
[13] X. Tian, L. Yang, J. Wang, Y. Yang, X. Wu, and X.-S. Hua, "Bayesian video search reranking," in *Proc. ACM Multimedia*, 2008.
[14] Y. Liu, T. Mei, X.-S. Hua, J. Tang, X. Wu, and S. Li, "Learning to video search rerank via pseudo preference feedback," in *Proc. ICME*, 2008.
[15] F. Schroff, A. Criminisi, and A. Zisserman, "Harvesting image databases from the web," in *Proc. ICCV*, 2007.
[16] J. Krapac, M. Allan, J. Verbeek, and F. Jurie, "Improving web image search results using query-relative classiÞers," in *Proc. CVPR*, 2010.
[17] K. Järvelin and J. Kekäläinen, "IR evaluation methods for retrieving highly relevant documents," in *Proc. SIGIR*, 2000.
[18] R. Fergus, P. Perona, and A. Zisserman, "A visual category Þlter for Google images," in *Proc. ECCV*, 2004.
[19] X. Tian, L. Yang, X. Wu, and X.-S. Hua, "Visual reranking with local learning consistency," in *Proc. MMM*, 2010.
[20] L. Wang, L. Yang, and X. Tian, "Query aware visual similarity propa-gation for image search reranking," in *Proc. ACM Multimedia*, 2009.
[21] X. Tian, D. Tao, X.-S. Hua, and X. Wu, "Active reranking for web image search," *IEEE Trans. Image Process.*, vol. 19, no. 3, pp. 805–820, Mar. 2010.
[22] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval, Cambridge Univ. Press, 2008. [Online]. Available: http://www.amazon.com/exec/obidos/redirect?tag=citeu-like07-20&path=ASIN/0521 865719.

[23] T.-Y. Liu, "Learning to rank for information retrieval," *Found. Trends Inf. Retr.*, vol. 3, pp. 225–33 1, Mar. 2009.
[24] T. Joachims, "Optimizing search engines using clickthrough data," in *Proc. KDD*, 2002.
[25] L. Wang, L. Yang, and X. Tian, "Query aware visual similarity propa-gation for image search reranking," in *Proc. ACM Multimedia*, 2009.

Author Profile

Sabitha M G received the B.Tech degree in Computer Science and Engineering from Mahathma Gandhi University in 2007. Currently she is doing M. E at Anna University Chennai in Computer Science and Engineering.As Part of M E Project for Web Image Retrieval using Reranking .