

Re-ranking of Web Image Search Results using a Graph Algorithm

Hilal Zitouni, Sare Sevil, Derya Ozkan, Pinar Duygulu
Bilkent University, Department of Computer Engineering, Ankara, Turkey
{zitouni, sareg, duygulu}@cs.bilkent.edu.tr, derya@usc.edu

Abstract

We propose a method to improve the results of image search engines on the Internet to satisfy users who desire to see relevant images in the first few pages. The method re-ranks the results of text based systems by incorporating visual similarity of the resulting images. We observe that, together with many unrelated ones, results of text based systems include a subset of correct images, and this set is, in general, the largest one which has the most similar images compared to other possible subsets. Based on this observation, we present similarities of all images in a graph structure, and find the densest component that corresponds to the largest set of most similar subset of images. Then, to re-rank the results, we give higher priority to the images in the densest component, and rank the others based on their similarities to the images in the densest component. The experiments are carried out on 18 category of images from [8].

1. Introduction

Web image search engines base their search algorithms on texts accompanying the images instead of the visual contents of these images. Therefore, results tend to provide significant number of images irrelevant to the desired query together with a small set of relevant images, since associated texts are generally irrelevant or misleading. On the other hand, performing a fully image based search by recognizing objects and scenes, is still not in the capability of computer vision systems.

Users want to see visually similar images corresponding to their query within the initial pages of the search results. Thus initiating from text based search results, a system that can list the visually relevant images in the first places and move the irrelevant images to the end, is likely to provide user satisfaction and be an alternative to visual based search engines.

Recently, some approaches are proposed in this di-

rection for re-organizing text based search results by incorporating visual information [1, 2, 4, 6, 8]. In [1], Ben-Haim et al. take a subset of images and segments all images into blobs. When clustered, densities of blob clusters become directly proportional to the relevancy of images in that cluster to the query. Using this idea, remaining images are inserted to appropriate clusters and images are re-ranked. Similarly, the objective of the work by Schroff et al. [8] is to form categorized image databases harvested from the web. The re-ranking operation is performed for separating relevant and irrelevant results by the usage of a combination of textual and visual features. In [6] a method that uses user intervention to re-rank the results is proposed. In their approach, a small subset of correct images is manually constructed and this subset is then used to filter the noise in the data. The method is applied to a set of musical instruments.

In this study, we propose a method to satisfy users of image search engines by re-ranking text based search results using visual information. We base our approach on the assumption that among the text based search results, there will be a subset which is visually relevant to the query and the images in this subset will form the largest group of images which are most similar to each other in the entire resulting set.

Based on this assumption, we find the largest subset of most similar images and list them with higher priority. Generalizing the method that we previously used to find the relevant faces associated with a name [7], in this study, we propose a graph based approach to find the group of relevant images in the result set of a text based search system.

In our approach, similarity of images are represented in a fully connected graph structure. With the use of this graph structure, the problem of finding the most similar group of images turns into the problem of finding the densest component in the graph. For this purpose we utilize the greedy densest component algorithm proposed by Charikar [3]. The images located in the densest component are assumed to be the relevant images, and are given higher priority. The rest of the images are

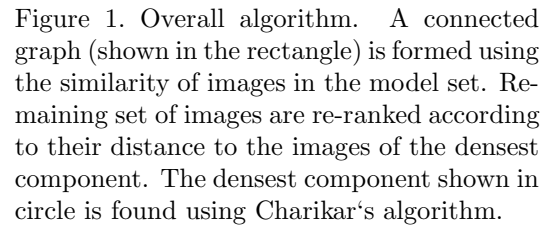
The following sections describe the details of the algorithm. We should note here that, the proposed algorithm only makes use of the available data, and does not require a supervised input to specify relevant images.

Our method consists of the following steps(see Figure 1): First, visual features are extracted from all images, and the similarity of each pair of images is computed based on the similarity of these features. Then, in order to reduce time complexity of the algorithm and increase the reliability, a subset from the first few pages of the original results is taken as a model set. In the fully connected graph constructed over this subset, the nodes are the images, and the edges are the similarity of images.

In the next step, the original graph with real valued edges is converted into a binary graph in order to apply the greedy densest component algorithm of Charikar [3]. Each time by removing a single node from the graph, the algorithm decides on the densest component of the graph, which has the largest number of nodes that are highly connected to each other.

2.1 Construction of the similarity graph

We first made use of SIFT operator proposed by Lowe [5] to detect and describe the interest points. However, rather than using the original matching criteria proposed in [5] which gives small number of matches due to large variety in web images, we propose a new matching scheme. Initially the Euclidean



distances between all interest points in pairs of images are computed. The best match for each point in an image is defined as the one with minimum Euclidean distance to the other image. This approach assigns a match to all of the points in the images. In order to eliminate the wrong matches, we apply a uniqueness constraint [7] which satisfies that there will be a unique, one-to-one match between the pairs of points, and the others not satisfying this condition will be eliminated. Once the matches are obtained, the similarity of two images is computed as the average distance of matching interest points. The fully connected similarity graph is constructed using these similarities, where each node represents an image and each edge represents the similarity between the nodes(images) it connects.

2.2 Forming the binary graph suitable for Charikar's algorithm

Charikar’s greedy densest component algorithm [3] requires a binary graph used for finding the most similar subset. As our fully connected similarity graph is a real-valued graph, a conversion is necessary. To do this, a threshold value was chosen, and the edges above that threshold was assigned the binary value 1, and the others to 0. The binary graph is constructed by keeping edges with value 1 and removing edges with value 0.

In the experiments, effects of different threshold values have been tested. It was found that these differences did not effect the results in a serious way; the choice of 0.3 is was observed as a satisfactory one. Once the binary graph is constructed, it is fed to the densest component algorithm.

2.3 Finding the densest component

Charikar’s algorithm for finding the densest component in a given boolean graph can be summarized as follows: a density value is computed for all subsets of the given graph obtained by removing one node each time. Then, the subset with the largest density is selected as the densest component. For a given subset S of the similarity graph, the density is computed as follows:

$$f(S) = \frac{|E(S)|}{|S|} \quad (1)$$

where $f(S)$ is the density of S , $|E(S)|$ is the number of edges in S , and $|S|$ is the number of nodes in S .

2.4 Expanding the model with other images

The densest component is found as described above on the subset of images which we refer as the model. Then, the remaining images are ranked according to their similarity to the elements of the densest component.

3. Experiments

In order to measure our performance we have used the data set provided by Schroff et al. [8] which consists of images harvested from Google’s image search results.

In the experiments, we have found that taking the first 30 images is generally sufficient as an initial model to capture the densest component. Once the initial model is formed, the remaining images are inserted to the model one by one, in accordance to their visual distance to the densest component. Thus, the densest component grows gradually. As it was explained in the previous section, after the complete similarity graph is constructed, images in the densest component are given higher priority and results are re-ranked accordingly. To visualize our performance we have plotted recall versus precision graphs for categories car and penguin in Figure 2. Because of the approach presented in [8] changes target set of images using additional elimination steps, we were not able to compare our performance to theirs. We were, however, able to compare our performance to

Google’s image search results as an example for web image search.

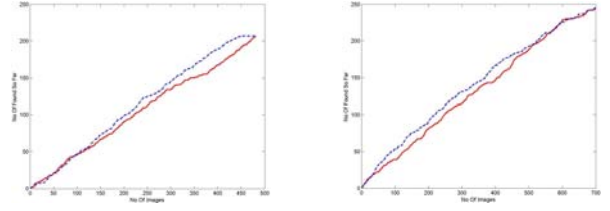


Figure 2. Precision vs Recall graphs for categories car(left) and penguin(right). Blue dashed line shows our performance; red solid line shows Google’s performance. Interest points were used for re-ranking.

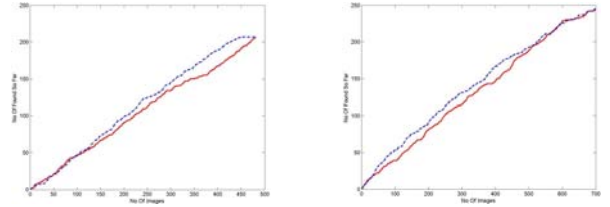


Figure 3. Precision vs Recall graphs for categories boat(left) and giraffe(right). Blue dashed line shows our performance; red solid line shows Google’s performance. Color features were used for re-ranking.

Table 1 shows the mAP values for our method compared to the ones of Google’s performance for the 18 different categories. Figures ?? and ?? visually show the rankings of Google and our approach for the first 20 images.

4. Summary and discussion

In this study, we propose a method to re-rank the Google’s image search results. The proposed method makes use of the observation that, although the original results include many irrelevant images, still the largest most similar subset of these images should correspond to the query. With the representation of image similarities in a graph structure, the problem have been converted into the finding of densest component in the graph.

The results are promising, with some categories producing better results than Google’s ranking. Although the mean average precision (mAP) values of the complete image sets do not appear to be high in some of the

Table 1. Mean Average Precision (mAP) values for interest point and color features of 18 categories.

Category	Google	Us	Color G.	Color Us
Penguin	0.4002	0.4397	0.4002	0.3576
Zebra	0.3842	0.4213	0.3842	0.3257
Car	0.4365	0.4661	0.4365	0.4390
Airplane	0.3555	0.3809	0.3555	0.3646
Guitar	0.5312	0.5437	0.5312	0.5289
Wristwatch	0.7777	0.7709	0.7777	0.7239
Camel	0.3786	0.3710	0.3786	0.3464
Elephant	0.4062	0.3982	0.3786	0.3464
Dolphin	0.4742	0.4615	0.4742	0.4680
Boat	0.4078	0.3894	0.4078	0.3820
Motorbikes	0.5700	0.5445	0.5700	0.5639
Bikes	0.4633	0.4271	0.4633	0.4727
Tiger	0.2494	0.2035	0.2494	0.2166
Kangaroo	0.3143	0.2537	0.3143	0.2503
Beaver	0.1616	0.0836	0.1616	0.1014
Shark	0.2869	0.1950	0.2869	0.2809
Horse	0.4220	0.3246	0.4220	0.4433
Giraffe	0.5716	0.4589	0.5716	0.5486

categories, the main objective of the approach, which is to provide users most relevant images at the top positions of the ranking is achieved in most of the cases.

Usage of different features effect the results. While interest point features give good results for categories like penguin, zebra, car, airplane and guitar, color features increase the performance of categories such as horse and bikes. Also, despite their overall results are not high, some categories like giraffe and boat have better rankings for the first 50 and 100 images when color features are used.

Low performance results are obtained due to two main reasons. First reason is that the proposed method assumes that in the model selected for finding the densest component, there are more instances of the query image compared to the others. When this is not the case, the proposed method forms an incorrect densest component. The second reason may be the visual features used the experiments. Since computing similarities is a crucial step in the approach, low performance in similarity computation effects the overall result. For some categories, especially with smooth surfaces, the interest point based matching does not provide a good similarity measure.



Figure 4. Ranking of ‘car’ query results. Google’s ranking is on the left; our ranking is on the right. Relevant images are in green boxes.

References

- [1] N. Ben-Haim, B. Babenko, and S. Belongie. Improving web based image search via content based clustering. In *IEEE Xplore SLAM*, New York City, NY, 2006.
- [2] T. L. Berg and D. A. Forsyth. Animals on the web. In *IEEE Conference on Computer Vision and Pattern Recognition*, New York City, NY, 2006.
- [3] M. Charikar. Greedy approximation algorithms for finding dense components in a graph. In *3rd International Workshop on Approximation Algorithms for Combinatorial Optimization*, London, UK, 2000.
- [4] W. H. Hsu, L. S. Kennedy, and S.-F. Chang. Video search reranking via information bottleneck principle. In *ACM Multimedia*, Santa Barbara, CA, 2006.
- [5] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004.
- [6] N. Morsillo, C. Pal, and R. Nelson. Mining the web for visual concepts. In *9th KDD Multimedia Data Mining workshop*, 2008.
- [7] D. Ozkan and P. Duygulu. A graph based approach for naming faces in news photos. In *IEEE Conference on Computer Vision and Pattern Recognition*, New York City, NY, 2006.
- [8] F. Schroff, A. Criminisi, and A. Zisserman. Improving web based image search via content based clustering. In *International Conference on Computer Vision*, Rio de Janeiro, Brazil, 2007.

5 Acknowledgments

This research is partially supported by TÜBİTAK Career grant number 104E065 and grant number 104E077.