Canonical Image Selection from the Web

Yushi Jing[1,2]
yjing@cc.gatech.edu

Shumeet Baluja[2]
shumeet@google.com

Henry Rowley[2]
har@google.com

College of Computing [1]
Georgia Institute of Technology
Atlanta, GA

Google Inc.[2]
1600 Amphitheater Parkway
Mountain View, CA

ABSTRACT

The vast majority of the features used in today's commercially deployed image search systems employ techniques that are largely indistinguishable from text-document search – the images returned in response to a query are based on the text of the web pages from which they are linked. Unfortunately, depending on the query type, the quality of this approach can be inconsistent. Several recent studies have demonstrated the effectiveness of using image features to refine search results. However, it is not clear whether (or how much) image-based approach can generalize to larger samples of web queries. Also, the previously used global features often only capture a small part of the image information, which in many cases does not correspond to the distinctive characteristics of the category. This paper explores the use of local features in the concrete task of finding the *single* canonical images for a collection of commonly searched-for products. Through large-scale user testing, the canonical images found by using only local image features significantly outperformed the top results from Yahoo, Microsoft and Google, highlighting the importance of having these image features as an integral part of future image search engines.

Categories and Subject Descriptors

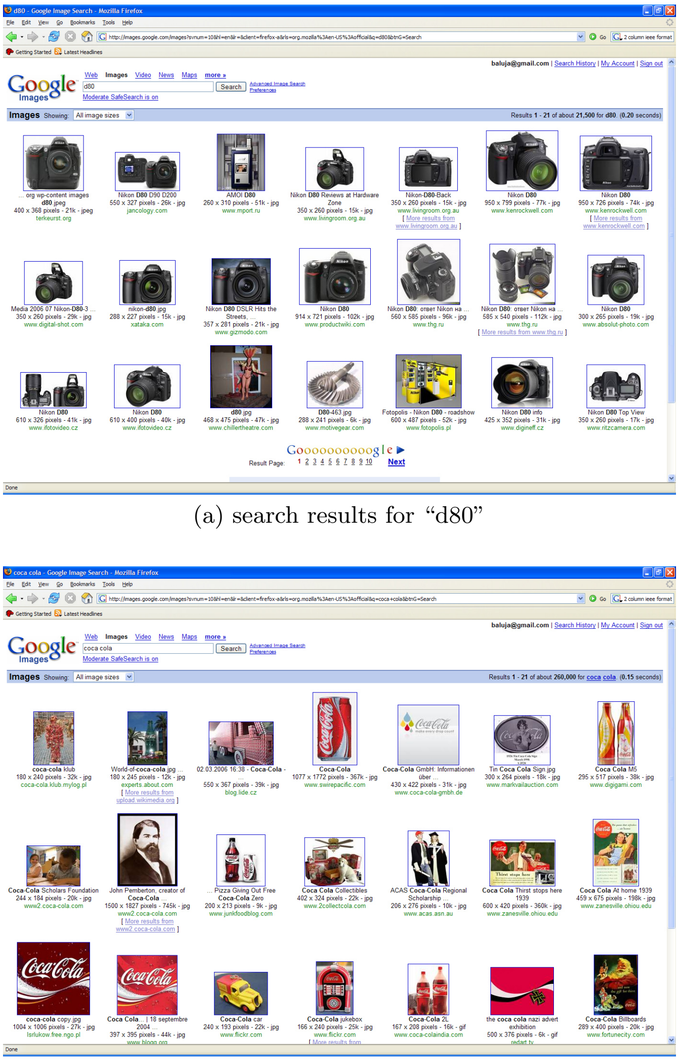
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.4.m [**Information System Applications**]: Miscellaneous

Keywords

web image retrieval, local features

1. INTRODUCTION

Although image search has become a popular feature in many search engines, including Yahoo, MSN, Google, etc., the majority of image searches use very little, if any, image information. Due to the success of text-based search of


d80 - Google Image Search - Mozilla Firefox
Google Images
Results 1 - 21 of about 21,500 for d80. (0.20 seconds)
Goooooooooogle
coca cola - Google Image Search - Mozilla Firefox
Results 1 - 21 of about 260,000 for coca cola. (0.15 seconds)

(a) search results for "d80"

(b) Search results for "coca-cola"

Figure 1: The query for "d80", a popular Nikon camera, returns good results on Google. However, the query for "Coca Cola" returns mixed results.

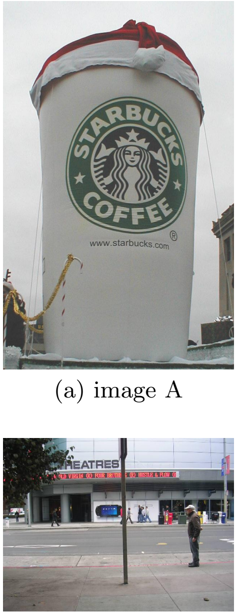
CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.
Copyright 2007 ACM 978-1-59593-733-9/07/0007 ...$5.00.

web pages, and in part to the difficulty and expense of using image-based signals, most search engines return images solely by examining the text of the pages from which the images are linked. For example, to find pictures of the Eiffel Tower, rather than examining the visual contents, images that occur on pages that contain the term "Eiffel Tower" are returned. No image analysis takes place to determine relevance or quality. This can yield results of inconsistent quality. For example, the query "d80", a popular Nikon camera, returns good results as shown in Figure 1(a). However, the query for "Coca Cola" returns mixed results as shown in Figure 1(b) - the expected logo or Coca Cola can/bottle is not seen until the 4th result. This is due in part to the difficulty in associating images with keywords, and in part to the large variations in image quality and user perceived semantic content.

Our approach relies on analyzing the distribution of visual similarities among the images. The intuition behind the approach is a simple premise: an author of a web page is likely to select images that, from his or her own perspective, are relevant to the topic. Rather than assuming that every user who has a web-page relevant to the query will link to a good image, our approach relies on the combined preference of many users. For example, in Figure 1(b), many of the images contain the familiar red Coca Cola logo. In some of the images, the logo is the main focus of the image, whereas in others it occupies only a small portion. Nonetheless, the fact that it is repeated in a large fraction of the images returned is an important signal that can be used to infer a common "visual theme" throughout the set. The images that best capture the visual themes found in the set of images should be returned to the user.

Content based image retrieval is an actively explored area and a number of overviews on this area have been published [4] [6] [13] [12]. The idea of analyzing the "coherence" of the top results from a traditional image search engine has been explored by Park et al. [11] and recently by Fergus et al. [3]. In particular, Park et al. [11] explored different clustering methods to select the top images from the initial search results. Our work is an logical extension to their works in the following three ways. First, as observed in [11] and [3], image features like color histograms and curvature only capture a small fraction of the information, which in many cases, does not correspond to the distinctive information of the category. For example as shown in Figure 2, when only color histogram are compared, image 2(a) is the closet match to image 2(b) out of the 1000 images returned by Google image search. On the other hand, local features are more robust against image deformation, variations and noise, and has demonstrated its potential for object category detection even with images collected from traditional search engines [2]. Therefore, we designed our experiments specifically to analyze the effectiveness of local features on potentially very noisy search engine image results.

Our second contribution is the scale of our experiments with human evaluators. [11] and [3] demonstrated the potential of "coherence" based image filtering, but it was not obvious whether (if so, how much) image-based system can improve the quality of search results when applied to a large set of queries, given the noisy nature of the web images and the rapidly improving image search engine results. We grounded our work on large-scale user evaluation (105 users) on a significant number of queries terms.



(a) image A

(b) image B

Figure 2: Given the 1000 images collected from Google search engine for the keyword "starbucks", image B is the closest match to image A when only color histogram is used.

The third are that that we differ from prior work is our novel and perhaps more difficult task. Our method attempts to find the *single* most representative image for popular product using only image features. We hope to quantify the potential performance improvement it can bring to the commercial search engines.

We chose to start with product searches (i.e. "ipod", "Coca Cola", "polo shirt", etc) for two reasons. First, this is an extremely popular category of searches, so improving this will affect many users. Second, it provides a good set of queries from which to quantitatively evaluate our performance: in many cases, there are only a few representative images of products that are acceptable. Our decision to examine the single most representative image was motivated by the importance and wide-applicability of this task. An enormous number of services, ranging from Froogle (Google's product search tool), NextTag.com, Shopping.com, to Amazon.com, all rely on being able to initially attract attention to products by showing a single image next to a product listing. We would like to automate the process of finding the most representative image for a product from the web. We quantitatively measure our results for this task by asking a group of users to compare the top result of our system with the top result returned by Yahoo.com, MSN.com and Google.com.

The remainder of the paper is organized into five sections. In the next section, we briefly review the underlying technologies used to determine the sub-image similarity. The reader who is familiar with SIFT features or similar local-descriptor based features may want to skip ahead to the following section. In Section 3, we describe how these features can be used for image-query results. Sections 4 and 5 describe how we measured the performance on the task and

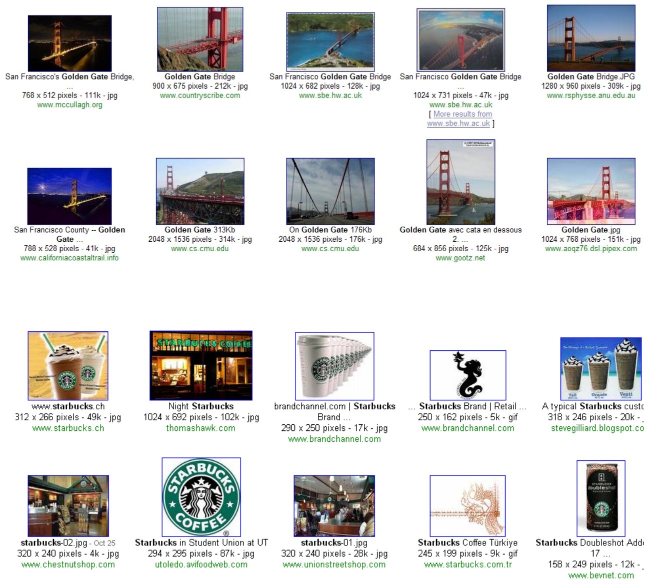

San Francisco's Golden Gate Bridge, ...
768 x 512 pixels - 111k - jpg
www.mccullagh.org
Golden Gate Bridge
900 x 675 pixels - 212k - jpg
www.countryscribe.com
San Francisco Golden Gate Bridge
1024 x 682 pixels - 128k - jpg
www.sbe.hw.ac.uk
San Francisco Golden Gate Bridge ...
1024 x 731 pixels - 47k - jpg
www.sbe.hw.ac.uk
[More results from www.sbe.hw.ac.uk]
Golden Gate Bridge.JPG
1280 x 960 pixels - 309k - jpg
www.rsphysse.anu.edu.au
San Francisco County -- Golden Gate ...
788 x 528 pixels - 41k - jpg
www.californiacoastaltrail.info
Golden Gate 313Kb
2048 x 1536 pixels - 314k - jpg
www.cs.cmu.edu
On Golden Gate 176Kb
2048 x 1536 pixels - 176k - jpg
www.cs.cmu.edu
Golden Gate avec cata en dessous 2. ...
684 x 856 pixels - 125k - jpg
www.gootz.net
Golden Gate.jpg
1024 x 768 pixels - 151k - jpg
www.aoqz76.dsl.pipex.com
www.starbucks.ch
312 x 266 pixels - 49k - jpg
www.starbucks.ch
Night Starbucks
1024 x 692 pixels - 102k - jpg
thomashawk.com
brandchannel.com | Starbucks Brand ...
290 x 250 pixels - 17k - jpg
www.brandchannel.com
... Starbucks Brand | Retail ...
250 x 162 pixels - 5k - gif
www.brandchannel.com
A typical Starbucks custo
318 x 246 pixels - 20k -
stevegilliard.blogspot.co
starbucks-02.jpg - Oct 25
320 x 240 pixels - 4k - jpg
www.chestnutshop.com
Starbucks in Student Union at UT
294 x 295 pixels - 87k - jpg
utoledo.avifoodweb.com
starbucks-01.jpg
320 x 240 pixels - 28k - jpg
www.unionstreetshop.com
Starbucks Coffee Türkiye
245 x 199 pixels - 9k - gif
www.starbucks.com.tr
Starbucks Doubleshot Add
17 ...
158 x 249 pixels - 12k -
www.bevnet.com

Figure 3: When a user queries on "golden gate" or "Starbucks", returned images are often taken from different locations, with different cameras, focal lengths, compositions, etc. Similarity measures for this task must handle potential rotation, scale and perspective transformations.

analyze the results of the user experiments. We also present an intuitive visualization method for users to quickly understand the content of the returned results. Finally, in Section 6, we close the paper with a summary of our findings and suggestions for future work.

2. COMPUTATION OF IMAGE FEATURES

The ability to identify similar sub-images is crucial to the performance of this application. As described in the previous section, global features like color histograms and shape analysis, when used alone, are too restrictive for our task. Instead, we select image local features that are rich in terms of local information content, yet stable under local and global perturbations in the image domain. Examples of local features include Harris corners [5], Scale Invariant Feature Transform (SIFT) [9], Shape Context [1], Spin Images [7] and etc. Mikolajczyk and Schmid [10] presented a comparative study of various descriptors and demonstrated experimentally that SIFT gives the best matching results. SIFT is widely popular in the computer vision community for its ability to generate highly distinctive features that are invariant to image transformations (translation, rotation, scaling) and robust to illumination variation. As a result, we chose to use SIFT features for our task.

The SIFT algorithm can be roughly divided into two stages: interest point selection and descriptor generation. Interest points refer to the locations in the image where visually distinctive features can be found. In addition to the x and y coordinates, SIFT also assigns a consistent scale and orientation to each interest point based on local image properties. Local features are then represented relative to their scale and orientation and therefore achieve invariance to resizing and rotation.

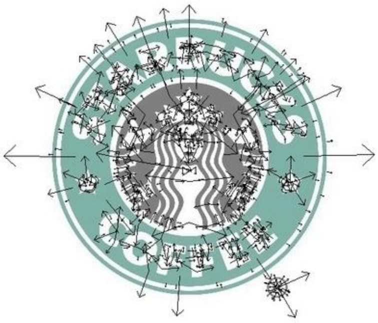


Figure 4: The interest points generated from starbucks image. The base, length and direction of each arrow corresponds to the x, y location, scale and orientation of the interest point.

Interest point selection is a three step process. First, SIFT builds a pyramid of scaled images by iteratively applying Gaussian filters to the original image. Next, adjacent Gaussian images are subtracted to create Difference of Gaussian (DoG) images, from which the characteristic scale associated with each interest points can be estimated by finding the local extrema over the scale space. Given the DoG image pyramid, SIFT selects interest points located at the local extrema of 2D image space and scale space. In the final step, we make the features invariant to rotation by assigning a characteristic orientation to each of the interest point. The interest points for the Starbucks logo are shown in Figure 4. The base, length and direction of each arrow corresponds to the x, y location, scale and orientation of the interest point. The scale and orientation attributes are clearly consistent with local image properties.

Given the interest points, SIFT generates distinctive yet stable descriptors for the pixels surrounding the interest points. A gradient map is computed for the region around the interest point and then divided into a collection of subregions, in which orientation histogram can be computed individually. In summary, this procedure has created a set of distinctive points by which to characterize the interesting characteristics of image. In the next section, we show how to use them for answering an image-search query.

3. CANONICAL IMAGE SELECTION

Since our method relies on the shared local properties of the web images, we call it "Local Coherence" (LC) selection method. We use a conventional image search engine, in this case Google image search, to generate up to 1000 initial image candidates, from which the most representative image is to be selected. In order to reduce the computational cost and avoid biasing towards larger images, all image candidates are resized to have a maximum dimension of 400 pixels. Each resized image contains 300 to 800 SIFT descriptors. Given these descriptors, our algorithm selects images that contain the most matching features.

Finding the nearest matches for roughly half a million high dimensional features can be computationally expensive, even when used in conjunction with an efficient data structure like metric trees. The recently proposed Spill Tree [8], an approximation to metric trees, is a more efficient alternative. For our task, we first construct a Spill Tree data

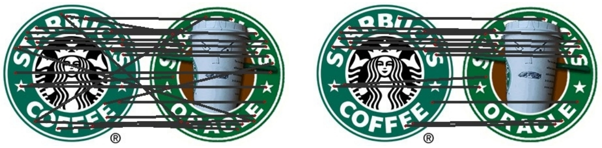


Figure 5: The gray lines represent matched feature points. Matches on the left are the results of individual feature matches. Matches on the right contain the remaining matches after common object verification.

Table 1: Local Coherence-based Image Selection Algorithm

1. Given a text query, retrieve the top 1000 images from Google image search and generate SIFT features for these images.
2. Identify matching features with Spill Tree algorithm.
3. Identify common regions shared between images by clustering the matched feature points.
4. Construct a similarity graph. If there is more than one cluster, select the best cluster based on its size and average similarity among the images.
5. From the chosen cluster, select the image with the most and highly connected edges.

structure from all the feature points in all images (roughly half a million total features). We then individually query each point in the data structure to find a collection of close matches. Two local features are considered as potential matches when their Euclidian distance is less than a threshold, defined loosely to allow for illumination changes and variations in image quality.

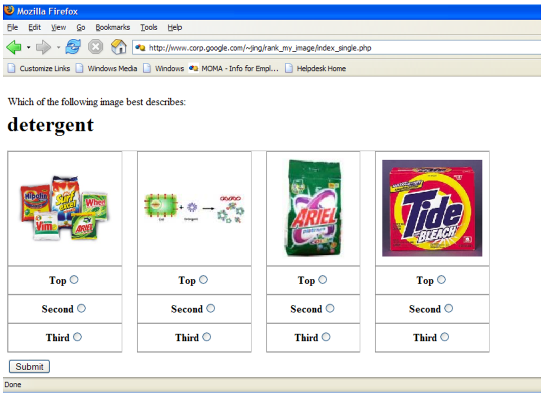
3.1 Common Object Verification

The loosely defined feature matching threshold allows for more correct feature matches, but it also introduces a significant number of false positives. Additionally, since visually similar local features can originate from different objects, feature matches alone can be semantically incorrect. Therefore, we apply clustering and geometric verification on the matched feature points.

We first group the matched points according to their corresponding image pairs, then a Hough Transform is used for object verification. This technique is commonly used in conjunction with local features [9]. A 4 dimensional histogram is used to store the "votes" the pose space (translation, scaling and rotation). At the end, we select the histogram entry with the most votes as the most consistent interpretation. For our task, we assume that there is only one instance of an object in any image, therefore only one consistent pose interpretation is selected. As shown in Figure 5, this procedure effectively rejects the erroneously matched feature points.

3.2 Image Selection

The previous step yields pairs of images that contain similar objects perhaps at different scales, taking up different portion of the image, with potentially different rotation, etc, from which similarity scores can be computed. In our work, we define the similarity score between two images as the number of matching points divided by their total number of interest points. This normalized similarity score has the additional advantage of favoring images with large and clear views of the object of interest.


Mozilla Firefox
File Edit View Go Bookmarks Tools Help
http://www.corp.google.com/~jing/rank_my_image/index_single.php
Customize Links
Windows Media
Windows
MOMA - Info for Empl...
Helpdesk Home
Submit
Done

Which of the following image best describes:

detergent

Top ○	Top ○	Top ○	Top ○
Second ○	Second ○	Second ○	Second ○
Third ○	Third ○	Third ○	Third ○

Figure 6: A screen shot of the experiment setup. The order of placement is randomly adjusted.

It is the easiest to think about the remainder of our algorithm in terms of a graph, with images as nodes, and their similarity scores (if larger than 0) as weighted edges. In this similarity graph, images that are not connected with any others are marked as outliers and removed. If there are multiple themes associated with a product (different product poses, etc), the resulting graph usually contains several distinctive clusters of images, as shown in Figure 9 [1]. For the task of generating the most representative image, we select the dominant cluster based on a combination of its size and average edge weights.

In the final step, we select the image within the cluster that is most heavily connected to other members in the cluster. If the similarity graph does not have a cluster, then we simply select the first image returned by Google as the best image. This can happen when the product itself lacks visually distinctive features (plastic bag, paperclip, etc), or if the object category is too vague or broad (pants, cars, etc). Table 1 provides a complete outline of the LC image selection algorithm.

4. EXPERIMENTS

The Local Coherence (LC) selection algorithm was used to select the most representative image for 130 product queries. The candidate images (up to 1000) are extracted from Google image search results on Nov. 6th, 2005. The selected images are be placed along with the top image from Yahoo, MSN and Google for user evaluation [2].

105 human evaluators participated in our experiment. Each evaluator was presented with 50 randomly selected sets of images, with the order of placement randomly adjusted. All images were resized to have a maximum dimension of 130 pixels. Figure 6 contains a snap-shot of the evaluation setup. The user was asked to give a strict ordering to the images

[1] For visualization purpose, we generated the maximum spanning tree from the similarity graph.

[2] The top image results from Yahoo and MSN were extracted on Aug. 10th 2006, with safe-image filter enabled.

Table 2: Number of votes received for each selection method. The overall score is computed as the weighted sum of all votes received (1st place votes * 3 + 2nd place votes * 2 + 3rd place votes). LC is able to identify common "themes" in 53 of the 130 product queries.

	Local Coherence	Google	Yahoo	MSN
53 product queries				
1st place votes	900 (43%)	344 (16%)	447 (21%)	389 (19%)
2nd place votes	579 (28%)	471 (23%)	544 (26%)	486 (23%)
3rd place votes	301 (14%)	661 (32%)	569 (27%)	549 (26%)
4th place votes	300 (14%)	604(29%)	520 (25%)	656 (31%)
Overall score (weighted)	4159 (33%)	2635 (21%)	2988 (24%)	2688 (22%)
130 product queries				
1st place votes	1686 (33%)	1090 (21%)	1466 (28%)	934 (18%)
2ndplace votes	1855 (35%)	1745 (34%)	761 (15%)	815 (16%)
3rd place votes	1057 (20%)	1469 (28%)	1499 (29%)	1151 (22%)
4th place votes	578 (11%)	872 (17%)	1450 (28%)	2276 (44%)
Overall score (weighted)	9825 (31%)	8253 (26%)	7419 (24%)	5583 (18%)

Table 3: Percentage improvement over competing systems.

	Google	Yahoo	MSN
53 products			
LC (1st place votes)	169%	105%	126%
LC (overall score)	58%	39%	55%
130 products			
LC (1st place votes)	57%	18%	83%
LC (overall score)	19%	32%	76%

by answering "which of the following image best describes" the text query. A new set of images was displayed after the user submitted their response. Although there was no time limit, the users usually finished the entire experiment in 10-20 minutes.

As described in the previous section, the LC selection algorithm uses the top result from Google image search as backup if it fails to find "common theme" among images. In this case, we display only three images (Google, Yahoo, MSN) to the evaluators. Given the user feedback, we assign a rank to LC either above or below (randomly selected to maintain fairness) the rank given to the Google image, and the adjust the ranks for other images accordingly.

Table 2 and Table 3 displays the aggregated evaluation results for the product images. We provide the voting results for all 130 product queries as well as for the 53 sets of products where a common theme is found by LC. We found the order in which the images are displayed does not affect how user rank the images in our experiment, with each position receiveing approximately 24% to 26% of votes.

5. ANALYSIS

As shown on Tables 2 and 3, LC significantly outperforms Google, Yahoo and MSN. Based on overall score, LC outperforms Google by 19%, Yahoo by 32%, and MSN by 76%. If the results were tabulated from the 53 sets of images where a common "theme" is found by LC, LC outperforms Google, Yahoo and MSN by 58%, 39%, 55% respectively. The improvement is even larger if we only consider the first place votes.

Figure 7 shows examples of competing images, sorted by the overall votes received by LC. A brief look at Figure 7 reveals that this performance improvement is not surprising. Many images selected by conventional search engines contain humorous or sexually explicit materials (which can be offensive for some people), as in the results of viagra, ipod, and rayban. Although perhaps popular, these images are not appropriate choice for product images, and certainly do not contain the common "theme" found in other images under the same query. On the other hand, some images selected by search engines are relevant and appropriate, but better choices are available. For example, both Google and Yahoo images for "Batman returns" are screen shots. Perhaps an image of the Batman DVD cover or movie posters would be a better choice. (The users did select the Batman DVD cover as a better choice.)

Local coherence algorithm is able to improve image selection by identifying the common "theme" in the initial image set, and select images containing the most visually distinctive representation of that theme. For example, Figure 8 contains the similarity graph for query "starbucks", "mona lisa" and "ipod." [3] The more connected images are evidently more relevant to the query and more visually appealing. Most of the off-topic or "funny" images are located as the leaves of the tree.

We think there are three reasons behind this result: First, people usually strive to take the best photos they can. Therefore we will see more frontal shots of the object than profile or occluded shots, and we will see more images under good lighting conditions, etc. Second, there is an implicit popularity contest for images on the web. Relevant and good quality photos tend to be repeatedly used for personal publishing (blogs, homepages) or selected manually as the featured image for online stores. For example in Figure 8(a), the Starbucks logo image are repeatedly used. This is important information for us to use. Third, images containing a dominant view of the object usually have more matches, therefore they have higher similarity scores than other images. This is crucial in selecting not only relevant, but also high quality images. For example, as shown in Figure 8(b), the original version of Mona Lisa image has the highest sim-

[3] For visualization purposes, we used a small number of image candidates, and extracted the maximum spanning tree from the similarity graph.

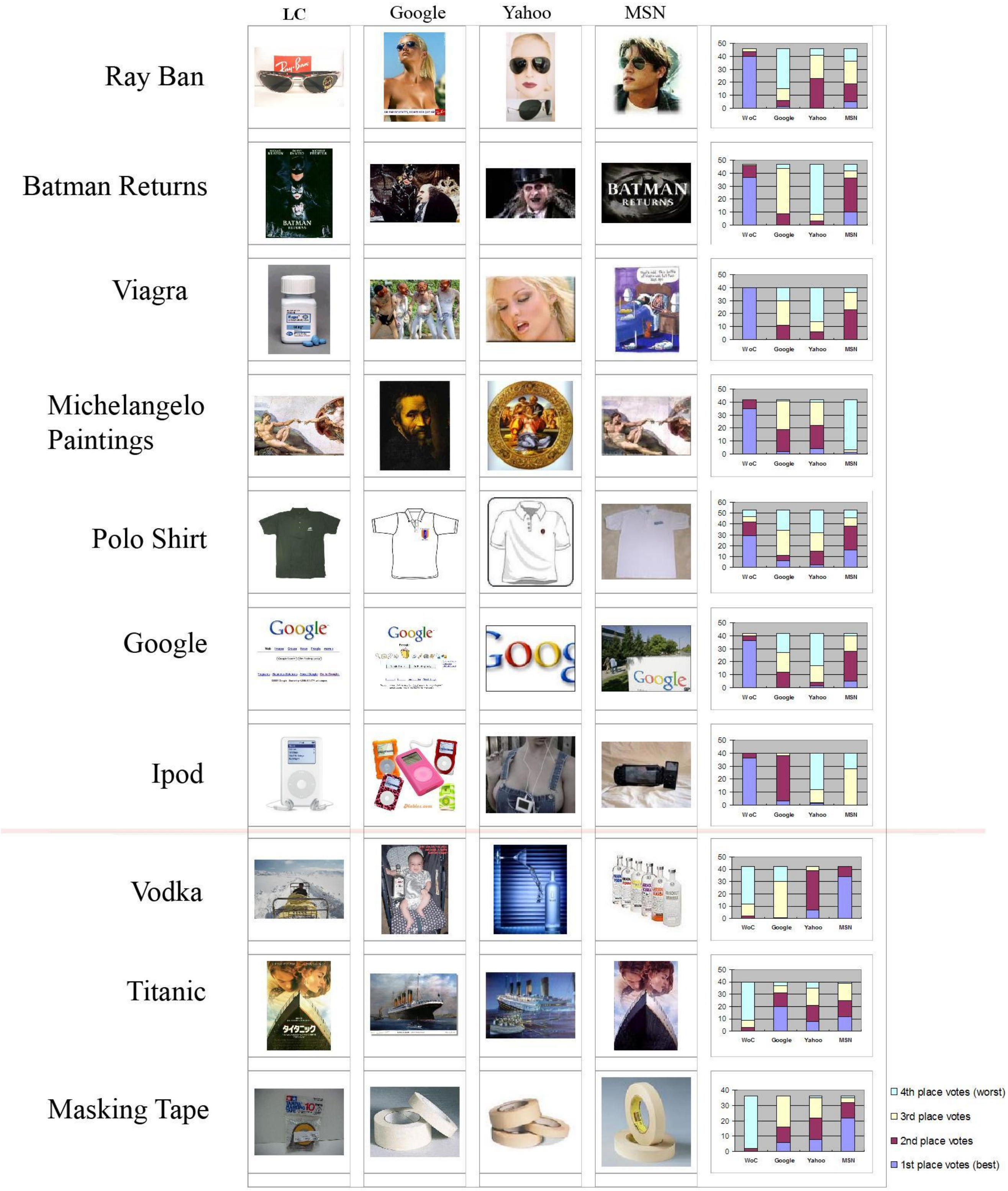
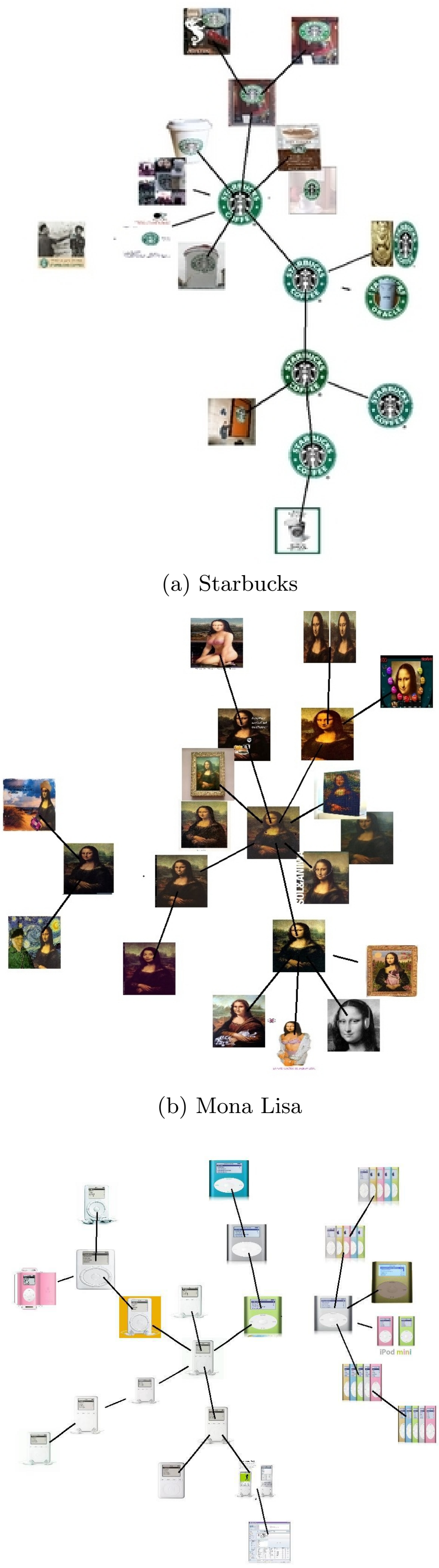


Figure 7: Examples of the 9 competing images, sorted by the overall votes LC received. (Images above the line are 6 queries that LC method does the best. Image below the line (3 images) are those LC method does the worst.) LC is the clear winner among the top 6 sets of images. "Masking tape" is an example where LC fails: a collection of similar (but not identical) images fooled LC into picking the bagged-tape as the best image.



(a) Starbucks

(b) Mona Lisa

(c) Ipod

Figure 8: Maximum spanning trees are extracted from the similarity graphs for visualization purpose. We selected the images in the center of each cluster as the representative images. As evidenced in the graph, the more connected images are more relevant to the query and are more visually appealing. Also, most of the off-topic or "funny" images are located as the leaves of the trees.

ilarity score with other images.

Another interesting observation is that, while LC selected images received 33%-43% of the first place votes, they received only 11% - 14% of the votes for being the worst image. This also shows that while our algorithm does not necessarily pick the most visually appealing image, the images selected are relatively relevant to the search query.

Currently our approach is computationally expensive due to the quadratic nature of our algorithm. Therefore, similarity measurements can only be generated off-line over a list of queries. In the future, we would like to explore methods to improve the training efficiency of our algorithm. Possible ways to reduce training time includes limiting the size of the image and the number of interest points, reducing the dimensions of local features, and use discriminative selecting features that are most related to the query we are interested in.

6. CONCLUSIONS AND FUTURE WORK

We have presented a method for selecting the best image among a group of images returned by a conventional text-based image search engine. The method exploits the local coherence implicitly present in the results by looking for an image which is the most representative of the group. An image is considered to be representative of the group if it is similar (as measured by local SIFT features) to many other images in the group. In a user study with product images, users ranked the image picked up the LC system first 43% of the time, compared with 16% for Google, 21% for Yahoo, and 19% for MSN's image.

There are a number of interesting directions for future work. First, and perhaps the most important, is expanding the range of queries for which the system is applicable. Local features like SIFT seem ideal for queries which have specific objects as results, like products. Queries for travel-related landmarks should also be handled well by this system. Further domains might require the use of other image features. For instance, face recognition methods may provide a useful similarity measure when a large portion of the image results contain faces. For queries where the results are an object category (eg "chair"), features typically used for content-based retrieval (color distributions, visual vocabularies) may be more fruitful.

The maximum spanning trees illustrated in Figures 8 and 9 contain a great deal of information to be exploited. The edges may be usable in the same way the web link structure is used to improve web page ranking. The arrangement of images may also be useful for visualizing the large result set in an intuitive way.

7. REFERENCES

[1] S. Belongie, H. Greenspan, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[2] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 2, pages 1816–1823, Oct. 2005.

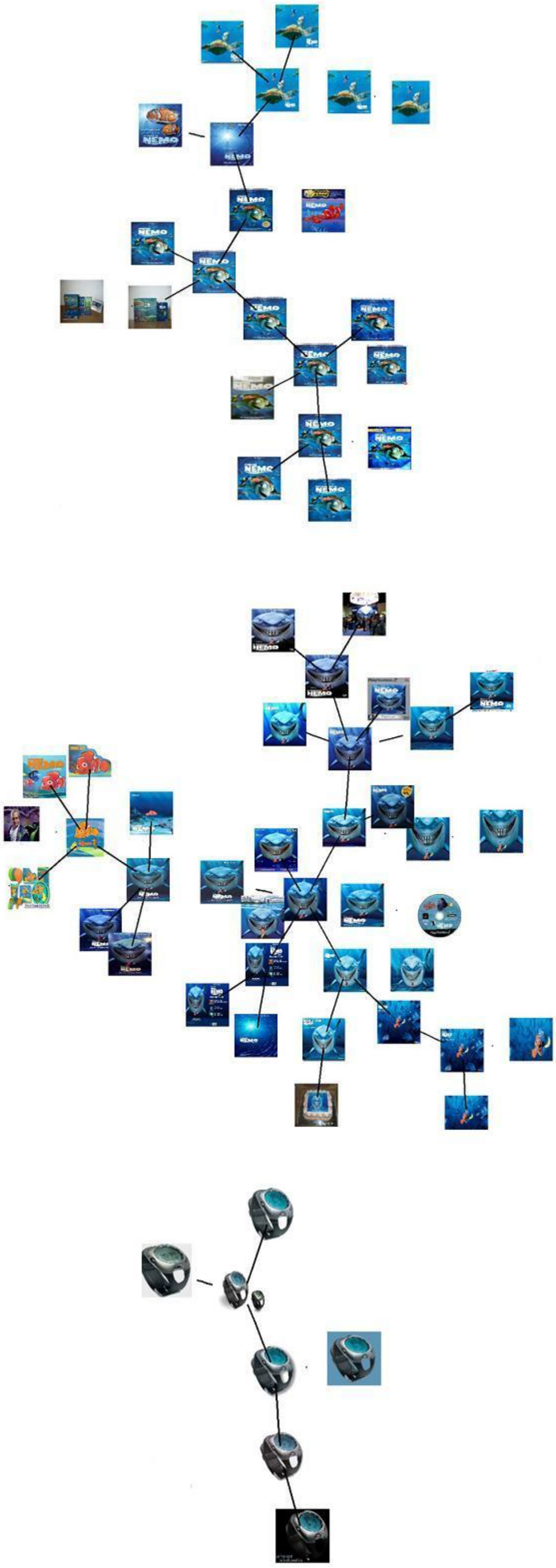


Figure 9: A collection of maximum spanning tree generated for the query "nemo." These clusters do not touch since they do not share any similar local features.

[3] R. Fergus, P. Perona, and A. Zisserman. A visual category filter for google images. In *Proceedings of the 8th European Conference on Computer Vision, Prague, Czech Republic*, pages 242–256, May 2004.

[4] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, 1995.

[5] C. Harris and M. Stephens. A Combined Corner and Edge Detector. In *PRoc. 4th ALVEY Vision Conference*, pages 147–151, 1988.

[6] R. Jain. Infoscopes, multimedia information systems. *Multimedia Systems and Techniques*, 1996.

[7] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using affine-invariant regions. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2003.*, pages 319–324, 2003.

[8] T. Liu, A. W. Moore, A. Gray, and K. Yang. An investigation of practical approximate nearest neighbor algorithms. In *Proc. 17th Conference on Advances in Neural Information Processing Systems (NIPS)*, pages 825–832, Cambridge, MA, 2005. MIT Press.

[9] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[10] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[11] G. Park, Y. Baek, and H. Lee. Majority based ranking approach in web image retrieval. pages 111–120, 2003.

[12] Y. Rui, T. Huang, and S. Chang. Image retrieval: current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, Apr. 1999.

[13] S. Santini and R. Jain. Similarity measures. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 21(9), 1999.