Search Web Images Using Objects, Backgrounds and Conditions

Jiemi Zhang Chenxia Wu Deng Cai

State Key Lab of CAD&CG, College of Computer Science
Zhejiang University, China
jmzhang10@gmail.com, chenxiawu@hotmail.com, dengcai@cad.zju.edu.cn

ABSTRACT

As the volumes of web images have grown rapidly in the last decade, Content-Based Image Retrieval (CBIR) has attracted substantial interests as an effective tool to manage the images. Most existing CBIR systems focus on the object in the image, while ignoring the conditions (day/night, sunny/rain, *etc*.) and the backgrounds, both of which are very helpful to meet the user's information need. To overcome this shortcoming, in this paper, we present a novel CBIR system depending on a novel query formulation considering three aspects: Object, Background and Condition. Specifically, we design a user-friendly interface to help the user formulate a query. The interface can allow the user to give the percentage, relative position and size of each object in the background. Moreover, a corresponding effective ranking method is proposed to return the desirable search results. Experimental results demonstrate that our proposed system improves the searching performance and the user experience compared with the existing searching systems.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Query formulation*

General Terms

Algorithms

Keywords

Content Based Image Retrieval, Query Formulation, User Experience

1. INTRODUCTION

The emergence of the World Wide Web has created many opportunities but also challenges for organizing and searching a large volume of images available publicly. Content-Based Image Retrieval (CBIR) has attracted substantial interests [1, 6, 7, 19, 11] as an effective technique to meet the users' increasing demands for searching web images.

MM'12, October 29–November 2, 2012, Nara, Japan.

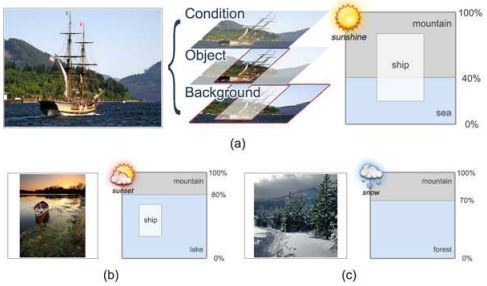

Figure 1: Examples of the OBC query formulation. (a) The image can be divided into three layers: the background layer (B: 50% mountain+50% river), the object layer (O: ship), and a global condition layer (C: sunshine) from the bottom to the top; (b) B: 20% mountain+80% river, O: boat, C: sunset; (c) B: 30% mountain+70% forest, O: Null, C: snow.

In the CBIR systems, the query formulation is the first and the key step to help the system return the desirable results to the users [2, 13, 4, 3, 9, 17]. A successful query formulation can not only provide a user-friendly interface to help the user conveniently describe the query but also let the system easily understand the users' requirement. We observe that the key concepts in images belong to three main categories:

1. Objects: Key objects in the image, such as *ship* in Fig. 1(a), which could be circled out with a rectangle. The size of the rectangle represents the object's proportion in the image.

2. Backgrounds: The background is often composed of one or two main scenes such as *sea* and *mountain* in Fig. 1(a). The boundaries of the two parts are often too complex to differentiate exactly and they are hard to be circled out on the image. Moreover, two kinds of scenes contribute different portion to the background. The exact percentage of each scene can be considered to satisfy the users' requirements.

3. Conditions: External conditions when shooting the picture, like seasons (spring or winter), weather (sunny or clouds), time of a day (day or night) (*sunny* in the above example). The most significant condition of one image is usually unique.

The traditional CBIR systems simply require the users to submit a visual-query, such as a sample image or a painted sketch [3]. To improve the query formulation, many efforts have been taken to focus on precisely describing the main object in the image [13, 9, 4]. Besides the objects, the background and the conditions (day/night, sunny/rain, *etc*.) of an image are also very helpful to return the

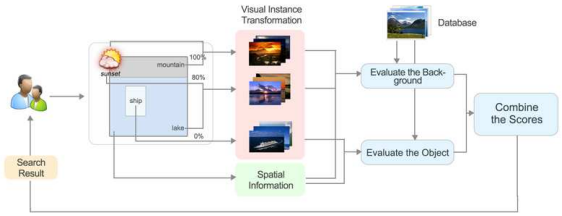

Figure 2: The flowchart of the proposed system.

desirable results. In the past, the background of an image is simply considered as a special kind of object [14, 16], which ignores its own characteristics. The weather (sunny/snow), time of a day (day/night), *etc*. which are the external conditions when shooting the picture are totally ignored in these systems. In practice, given the percentage of each scene in the background, the user can easily describe their requirements especially when making slides or designing a pop. And the searching results may be totally different for the same objects and background with different proportions.

To overcome these limitations, in this paper, we design a novel system based on a novel query formulation using Objects, Backgrounds, Conditions, called *OBC query*. Compared with the existing CBIR systems, the novel user interface (illustrated in Fig.3) is easier to describe the users' demands. A query can be described by simply drawing the Object regions, sliding the Background elements with the proportions, selecting or inputting the type of Condition. With the OBC query, our system can easily understand the main object, the background and the condition in the query; the relative size of the object to the background; and the proportion of each element in the Background. To make fully usage of these information, we propose a corresponding ranking method. The OBC query is first translated into visual instances and spatial information. Then the background and the condition are merged together to evaluate the relevance, meanwhile the object is evaluated separately. Finally we combine the scores with an uniform formulation to rank the images in the database. Experimental results show that our system enhances the searching performance and the user experience compared with the existing searching systems.

2. APPROACH

Fig. 2 shows the main flowchart of our system. The searching process can be divided into three steps: 1) the users submit an OBC query; 2) the system translates the OBC query into the visual instances and the spatial information; 3) the system evaluates and ranks the relevance of the images in the database.

2.1 Query Formulation

Fig. 3 is a snapshot of our system's user interface. To formulate an OBC query, the user first chooses a desired condition (or input one in the text-box), type one or two keywords of the background and dragging the separating line between them to determine the percentage of these two scenes, and draw a rectangle with the keywords to represent the object. It is also ok if the user only provides one or two aspects of the object, the background and the condition. To help the user describe the query, our system also provides some template queries under the canvas. In summary, the user only needs to edit some keywords or drag the separating line to submit a comprehensive OBC query in our system.

2.2 OBC query translation

Once the user submit an OBC query Q, the system first translates it into the visual instances of keywords and the desired distribution of each keyword in the query.

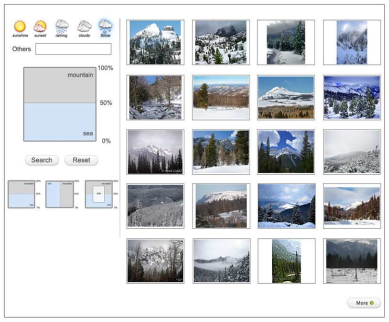

Figure 3: User interface of the proposed system. The example describes the image with half "forest" and half "mountain" from top to bottom in a snowing day.

Visual Instance Translation The *visual instances* are several representative images that can represent a certain keyword. Translating the keywords to visual instances can effectively bridge the semantic gap [17, 5]. In our system, we first use the keywords submitted by the user (either "condition+one part of the background" or "the object") as a text query to collect a group (with the size 50) of web images from the text-based search engine as the candidate set. We call these two types of keywords as *background keywords* and *object keywords* respectively. Then we use the affinity propagation (AP) algorithm [5] to find m most relevant images from the candidate set. For the background keywords, we exactly use these m images as the visual instances. While for the object keywords, we use the salient objects detected from the images by a learning-based approach [10] as the visual instances. After this step, all the keywords in the query are represented by visual instances so that we can ranking the images in the database by computing their similarity to the visual instances.

Spatial Translation The visual instances learned in the previous step only represent the keywords of the OBC query. To utilize the spatial information, we need a *spatial translation* step. Most of the images contain less than two main scenes in the background and both the relative position and the proportion of each part in the background can be acquired directly. Specifically, two main types of the relative position between two parts are considered: the left-right ($l-r$) and the up-down ($u-d$) and the proportion p_{b_l}, p_{b_r} or p_{b_u}, p_{b_d} of these two parts can be obtained from the separating line directly. For simplicity, we represent the percentage as $p_Q = p_{b_l}/(p_{b_l}+p_{b_r})$ or $p_Q = p_{b_u}/(p_{b_u}+p_{b_d})$.

For the object keywords, similar to [17], we use a 2D Gaussian distribution $G(x,y)$ to estimate its position and size, which is decided by the mean $\mu_k = [x,y]^T$ and the covariance matrix $\Sigma = Diag((\theta w)^2, (\theta h)^2)$. Here (x,y) is the the center of the rectangle and h, w are its height, width. The θ is a constant setting to be $\sqrt{(2log(2))^{-1}}$. Thus, the distribution degrades to a half near the boundary of the rectangle. The shape of the distribution is determined by the size of the rectangle: the smaller the rectangle is, the more rapidly the distribution degrades to zero from the center.

2.3 Ranking the Database

After translating the OBC query Q to visual instances and spatial information. In this step, our goal is to find a set of relevant images $\mathcal{S}_\mathcal{Q}$ from the whole database $\mathcal{S}$ for the given query Q. The images $q \in \mathcal{S}_\mathcal{Q}$ should not only include the keywords mentioned in the query, but also conform to the spatial constraints.

To represent the images, We use the Bag-of-Words (BoW) model, which is the state-of-the-art method for image representation. In our system, we split each image c in the database into $n \times n$ patches and then extract visual features (the SIFT feature, the color fea-

Figure 4: From left to right, the four images show: (1) the original picture; (2) the real separating line (red line) of the two parts of the Background; (3) the alternative separating lines (white line); (4) the best separating line choose by the system with the $Sepp$ score.

ture, *etc*.) from each of the patch independently. The BoW feature vector is then calculated for each patch. In this way, the image c is represented as $\{\mathbf{f}_{c_1,c_2,\ldots,c_{n\times n}}\}$, where $\mathbf{f}_{c_i}(i = 1, 2, \ldots, n\times n)$ is the BoW feature of the patch c_i. Similarly, m visual instances for the k-th keyword of the query can be represented as $\{\mathbf{f}^k_{v_i}\}(i = 1, 2, \ldots, m)$ using the BoW feature.

2.3.1 Computing the content relevance

To evaluate the relevance score between the keyword k and the image c , we summate the similarities between $\mathbf{f}_c$ and each visual instance for the k-th keyword $\mathbf{f}^k_{v_i}$:

$$\alpha(k, c) = \sum_i Sim(\mathbf{f}^k_{v_i}, \mathbf{f}_c),\ Sim(\mathbf{f}^k_{v_i}, \mathbf{f}_c) = \sum_j \frac{\langle \mathbf{f}^k_{v_i}, \mathbf{f}_{c_j} \rangle}{|\mathbf{f}^k_{v_i}||\mathbf{f}_{c_j}|}$$

where $\langle , \rangle$ denotes the dot product of two vectors.

2.3.2 Integrating the spatial information

We introduce how we evaluate the relevance score of the background keywords (denoted as β_B) in this section. The relevance score of the object keywords (denoted as β_O) is evaluated using the method in [17]. As mentioned above, we only consider two main types of relative positions for the background: the left-right ($l - r$) and the up-down ($u - d$). Here we give the up-down as an example, and the left-right could be evaluated in the same way.

As Fig. 4 shows, for an OBC query with the background which is compose of 'up mountain, down river', there is an irregular but explicit separating line between the two main scenes described by two background keywords denoted by b_u, b_d for up and down. In other words, the separating line split the image c into two parts with two scenes: the up part (c_u) and the down part (c_d). As defined in Eqn. 1, $\alpha(b_u, c_u)$ measures the relevance between the keyword b_u and the up part of image, which is similar for $\alpha(b_u, c_d)$, $\alpha(b_d, c_u)$ and $\alpha(b_d, c_d)$. Then the relevance score $Sepp$ to measure the relevance of the background keyword and the corresponding part asides the separating line l can be defined as $Sepp(l) = (\alpha(b_u, c_u) + \alpha(b_d, c_d)) - \lambda|\alpha(b_u, c_u) - \alpha(b_d, c_d)|$, where $(\alpha(b_u, c_u) + \alpha(b_d, c_d))$ measures the similarity between the right corresponding pair of the keyword and the image part. The penalty $|\alpha(b_u, c_u) - \alpha(b_d, c_d)|$ is added to those images which only has a very high score on one side and a low score on the other side. λ is a positive parameter.

As each image is split into $n\times n$ patches, there are n possible separating lines. For each image $c \in \mathcal{S}$, we define $Sepp^*(c) = max_i Sepp(l_i)$ as the relevance score of the background keywords with the most suitable separating line l^*. l^* split the image into the up part c^*_u and the down part c^*_d.

Note that above relevance score definition does not consider the obscureness by the front object. To reduce this influence, we refine the relevance score by redefining the $\alpha(b_u, c^*_u)$ and $\alpha(b_d, c^*_d)$ by considering the object keyword o:

$$\alpha(b_u, c^*_u) = \begin{cases} \alpha(b_u, c^*_u) + \alpha(o, c^*_u) & \text{if } \alpha(o, c^*_u) > 0 \\ \alpha(b_u, c_u) & \text{otherwise} \end{cases} \quad (1)$$

To evaluate the percentage of the image c, we count the number of the patches (n_u and n_d) which contain the corresponding keywords ($\alpha^*(b_u, c_i) > 0, \alpha^*(b_d, c_i) > 0$). Then the percentage could be calculated as $p_c = n_u/(n_u + n_d)$. Finally we evaluate the relevance score of the background keywords as $\beta_B = Sepp^*(c) * (1 - |p_Q - p_c|)$, where p_Q is the user's given percentage obtained in the spatial translation.

2.3.3 Combining the Relevance Scores

To rank the images in the database, we combine the relevance scores of both the background keywords and the object keywords. Typically, we combine them with an uniform formula:

$$r = E(\beta) - \frac{\gamma}{2}[|\beta_B - E(\beta)| + |\beta_O - E(\beta)|],\ E(\beta) = \frac{1}{2}(\beta_B + \beta_O)$$

3. EXPERIMENTS

In this section, we evaluate the performance of our proposed system compared with state-of-the-art methods on extensive collected web images. We use 50 query tasks to evaluate the searching performance. 10 of them include two of the object, the background and the condition and the remaining 20 include all three aspects. 5 Conditions (sunshine, sunset, night, snow and cloud), 6 scenes (sea, mountain, buildings, forest, desert, street, thus we have about ten mixed backgrounds) and 4 objects (ship, car, flowers and chair) are evaluated in our experiments. Specifically, for a given query task, the image database is obtained by querying a text-based image retrieval engine with the keywords of the task. All possible combinations of keywords are used as the text query, and the top 500 images of the results compose the whole database. Some images are from the COREL image collection and some are collected from the Internet, including Google[1], Flickr[2], and [15].

We take the text-based image search system as the baseline. To accomplish a task with a text-based image search system, the keywords of the task are used to query the system. Taking the task of Fig. 1(a) as an example, the query would be "mountain sea sunshine ship", and the order of this query would be randomly. Moreover, the noisy free-text query is not adopted in the experiment. We also involve the "show similar image" search in the comparisons. This method describes a class of images by an sample picture. To accomplish a task with the method of "show similar image", we collect the most relevant images of the top 10 images from the baseline system and use them as the sample picture one by one. The performance is evaluated by averaging the 10 searching results.

For our proposed system, the OBC query is used as introduced in Section 2.1. We adopted two kinds of visual features: SIFT feature [12] and color feature. Given a query, we select which feature to use adaptively to measure the condition, exploiting query classification method in [18]. The SIFT feature is also used to evaluate the spatial information of the background and the object. The number of the visual instances for one keyword m is set to 5. λ is set to 0.1 and γ is set to 0.8 heuristically.

To perform quantitative evaluation, some volunteers are recruited to label the ground truth. For a given task, an image is labeled with a relevance score, according to how well the image accords with the search intention. To differentiate the relevance degrees, the relevance score is defined in four levels from level 0 to level 3 [17]. Level 3 corresponds to the most relevant (the image is consistent with the description of the query), and level 0 the least relevant (the condition dose not match and some parts of the background or the object in the query is missing). Normalized Discounted Cumulative

[1]http://images.google.com/
[2]http://www.flickr.com/

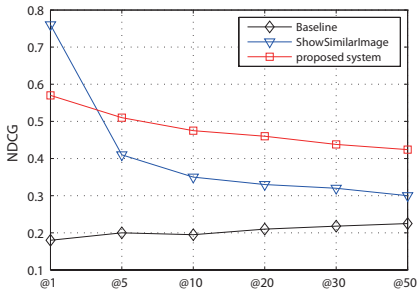

Figure 5: Quantitative results of three compared methods.

Gain (NDCG) [8] is adopted to measure the image search performance. The average NDCG scores are calculated in all query tasks to numerically compare the methods.

3.1 Results

Fig. 5 shows the quantitative results of three compared methods. It is obvious that the proposed system shows a satisfactory performance. It outperforms the baseline remarkably in all NDCG depth and is superior to "show similar image" method except of NDCG @1. This means the search intention is well interpreted by the proposed system through the proposed OBC query. However, the text-based images search engine performs poorly, because it does not consider the spatial information and the proportion. The method of "Show Similar Image" gets a very high NDCG score @1, it is mainly because the returned image at the first rank is supposed to be the sample itself. However, the performance drops rapidly along with growth of the NDCG depth. It is because visually similar images are not ensured to have the same semantic concepts, but only focus on the color, the texture, or the unclear tags around the image.

We also present the visual results to visually show the advance of our system. We give search results between the traditional text-based search engine and our system for three query tasks including the mixed background and object. Three text-queries are used to test the text-based engine and their results are shown in the left part of Fig. 6. Since the text-based engine does not consider the proportion of the background, the results returns randomly. In the right part of Fig. 6, we show the corresponding search results of the proposed system. It can be seen that most images returned by our system are consistent with the users' requirement.

4. CONCLUSIONS

In this paper, we propose a novel content-based image retrieval system, which searches the images by Object, Background and Condition. Different from traditional image search systems which focus on the appearance of the object, our system concerns the condition, the proportion of elements in the background, and the spatial information of the object. A user-friendly interface was designed and a corresponding ranking algorithm was developed. Experiments have shown that the proposed system can return more desirable results and provide more satisfactory user experience compared with the existing searching systems.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China under Grants 91120302 and 60905001, National Basic Research Program of China (973 Program) under Grant 2011CB302206.

5. REFERENCES

Figure 6: Visual results for three different queries "sea+tall buildings+sunshine", "mountain+sea+ship+sunshine" and "mountain+street+car+sunset" in the middle. The left side is the results of the traditional text-based search engine, and the right side is our system's result.

[1] D. Cai, X. He, and J. Han. Spectral regression: A unified subspace learning framework for content-based image retrieval. In *Proceedings of the 15th ACM International Conference on Multimedia*, pages 403–412, 2007.
[2] D. Cai, X. He, W.-Y. Ma, J.-R. Wen, and H. Zhang. Organizing WWW images based on the analysis of page layout and web link structure. In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo*, pages 113–116, 2004.
[3] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu. Sketch2photo: Internet image montage. *ACM Transactions on Graphics*, 28(5), 2009.
[4] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, 2007.
[5] D. Dueck and B. J. Frey. Non-metric affinity propagation for unsupervised image categorization. In *ICCV*, 2007.
[6] X. He, D. Cai, J.-R. Wen, W.-Y. Ma, and H.-J. Zhang. Clustering and searching www images using link and page layout analysis. *ACM Transactions on Multimedia Computing, Communications and Applications*, 3(1), 2007.
[7] X. He, W. Min, D. Cai, and K. Zhou. Laplacian optimal design for image retrieval. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 119–126, 2007.
[8] K. Jarvelin and J. Kekalainen. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR*, 2000.
[9] E. P. X. Li-Jia Li, Hao Su and L. Fei-Fei. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*, 2010.
[10] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(2), 2011.
[11] Y. Liu, T. Mei, and X.-S. Hua. Crowdreranking: exploring multiple search engines for visual search reranking. In *SIGIR*, 2009.
[12] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 60(1), 2004.
[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.
[14] A. Quattoni and A. Torralba. Recognizing indoor scenes. *CVPR*, 2009.
[15] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. *IJCV*, 77(1-3), 2008.
[16] C. Schmid. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.
[17] H. Xu, J. Wang, X.-S. Hua, and S. Li. Image search by concept map. In *SIGIR*, 2010.
[18] R. Yan. Learning query-class dependent weights in automatic video retrieval. In *ACM Multimedia*, 2004.
[19] Z.-J. Zha, L. Yang, T. Mei, M. Wang, and Z. Wang. Visual query suggestion. In *ACM Multimedia*, 2009.