Combining Visual Exploration and Searching for Interactive Texture Retrieval

Bart Thomee Mark. J. Huiskes Erwin Bakker Michael S. Lew

LIACS Media Lab, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, Netherlands

Abstract

We propose an interactive technique that allows the user to visually explore the feature space around relevant images and to focus the search on only those regions in feature space that are relevant. Through the use of a novel interface, the user can adjust the exploration front around each relevant image, visually setting the range within which images are also considered to be relevant. By giving feedback, the relevant nearest neighbors at the frontier can be further explored and any non-relevant ones removed, resulting in a more optimally refined search space.

1 Introduction

Over the course of years, research on content-based image retrieval has received increasing attention, and retrieval techniques have evolved from basic methods operating in low-level feature space into advanced approaches that often incorporate mechanisms from other disciplines, e.g. genetic algorithms [13]. This diversity in techniques notwithstanding, the general consensus is that the application of relevance feedback leads to improved search results and therefore has been applied in the majority of research from the moment the concept was introduced by Rocchio [1] in 1971.

The current trends in image retrieval focus away from operating directly in low-level feature space and attempt to approach image retrieval from a higher level, for instance by discovering subspaces or manifolds in feature space where similar images reside (e.g. [2], [3]) or at an even higher level by focusing on core image concepts and semantics (e.g. [4], [5]). The belief in the power of popular opinion has sparked interest in long-term learning (e.g. [6], [7]), where the idea is that an image considered to be relevant by several users for a particular search query is likely to be relevant as well for another user if the same or a similar query is performed.

Although the relevance feedback algorithms have received significant attention in the research literature, relatively little effort has been directed at new user interfaces for aiding in the search process. One of the grand challenges in our field is considered to be the need for experiential exploration systems that allow the user to gain insight into and support exploration of media collections [18]. For users, exploration is the predominant mode of interaction, rather than querying, and therefore interfaces that accommodate for this behavior are needed [17]. In this paper we therefore propose a novel interactive technique that allows the user to visually explore the feature space around relevant images and to focus the search on only those regions that are relevant. Each of these regions centers on a relevant example image and is bounded by its relevant nearest neighbors.

In Section 2 we will look at related work and in Section 3 we will discuss the proposed exploration technique in detail. Section 4 describes the experiments we performed and we conclude in Section 5.

2 Related Work

When it comes to visualizing the image collection, most research focuses on how to present the search results [14], whereas work on visualization for assisting the user to search more efficiently is rather limited. In [15] a similarity-based visualization technique is used to project the image collection onto a

2D manipulation space, where the user can easily select groups of similar images together and refine the selection. Hyperbolic visualization of a concept ontology is used in [4], allowing the user to obtain an overview of the image collection at a concept level and to interactively navigate the concept ontology by zooming in on different concepts of interest. The notion of visual islands is introduced in [16] to fulfill the principal goal of guided user browsing. This includes a process called island hopping that is used to dynamically reorganize the displayed pages according to the user's selection, so that the user can explore deeper into a particular dimension that he or she is interested in.

3 Exploring Feature Space

Unlike in semantic spaces, where ideally all images of interest are clustered together in a certain area, in low-level feature space these images can be spread out over multiple areas. For example in a feature space built up on color features, it is likely that images of differently colored tulips can be found in several parts of the space. Such a search can be performed using multiple query points (e.g. [8], [9]), but exploring the feature space around each query point is often a slow process. Most interfaces only present a limited number of images to the user, putting a heavy burden on the user as navigating the space around each query point requires many iterations of feedback.

To reduce user effort and allow efficient refinement of the relevant search space, we propose a novel technique where the span of the search space surrounding relevant images is visualized and can be interactively adjusted by the user. In Figure 1 is illustrated how the user establishes the search space by exploring the nearest neighbors of relevant images. Initially, e.g. from a random selection of images from the database, the user marks one or more images as relevant and then for each of them proceeds to explore its nearest neighbors in feature space, increasing the number until on the border non-relevant images appear. In subsequent steps, the search space can be refined by removing non-relevant images and exploring relevant nearest neighbors.

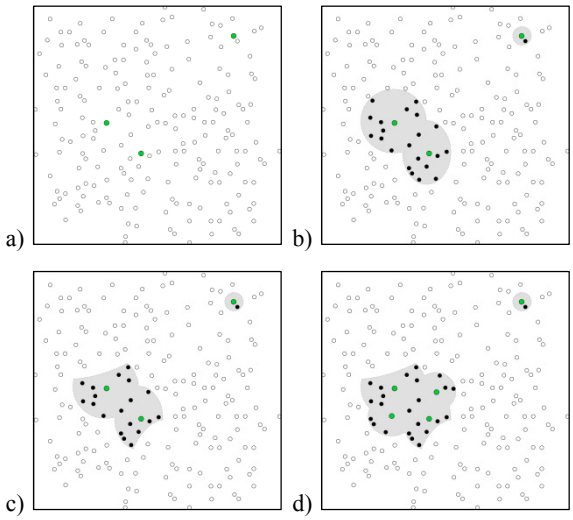

Figure 1. Establishing the search space: a) Three relevant images are indicated by the user, shown as larger dots with a green center. b) The user expands the search space around the relevant images to define suitable regions containing a high percentage of relevant nearest neighbors. c) The search space is contracted by the user by removing two non-relevant nearest neighbors that are located on the border. d) Two other nearest neighbors are explored and the search space is expanded with their relevant nearest neighbors.

3.1 Visualization and Interaction

The user can explore an image from within our interface, as is shown in Figure 2. Initially only the selected image is displayed, but by adjusting the exploration front more and more of its nearest neighbors are shown. The user decides where the exploration front lies. In case a positive feedback image is

explored, the exploration range ideally encompasses a collection of nearest neighbors of which a high percentage is considered to be relevant. In case a negative feedback image is explored, the exploration range ideally contains a collection of highly non-relevant images. In both situations, the small number of non-relevant images present in the collection of relevant images, or vice versa, can be removed by the user at a later stage. When the number of explored nearest neighbors becomes too large to be displayed in a comprehensible manner, a random selection is made. Yet, all nearest neighbors that fall within the exploration range are used in the construction of the relevant search space.

The retrieval system collects the positive feedback and negative feedback images and their explored nearest neighbors. Then a new iteration is started where the user is presented with the most informative images, which are those images that have the highest information scores, and with the best images, which are those images that have the highest relevance scores. The user can continue refining and exploring the search space until he or she is satisfied.

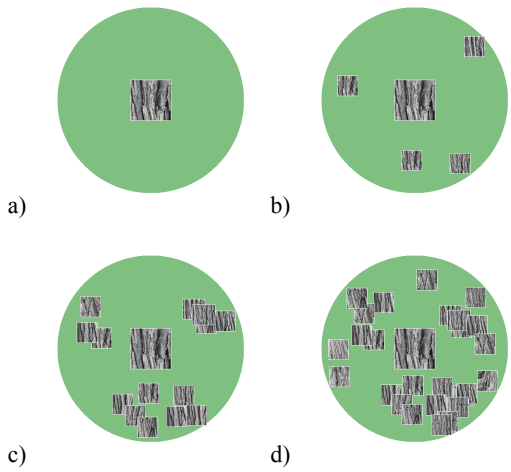

Figure 2. Exploring feature space: a) Initially only the positive feedback image is shown in the center. b-d) The user expands the search range several times until one or more non-relevant nearest neighbors appear on the border. The distance of an image to the center depends on the distance measure used by the retrieval system. A small distance in feature space places the image near the center, whereas a large distance places the image near the edge.

3.2 Feedback Sets

Let the positive feedback example set S_t^+ at iteration t consist of all selected relevant images gathered thus far

$$S_t^+ = \{(s_1^+, r_1^+), \cdots, (s_{n_t}^+, r_{n_t}^+)\} \ , \tag{1}$$

where, for each example image s_i^+, r_i^+ is the exploration range as selected by the user. The negative feedback example set S_t^- is defined similarly.

Let A_{ti}^+ be the set of images at iteration t within the exploration range of a positive feedback example

$$A_{ti}^+ = \{x \in D | d_w(s_i^+, x) < r_i^+\} \ , \tag{2}$$

where x is an image from the image database and (s_i^+, r_i^+) are from the i-th tuple of S_t^+. Let A_{ti}^- be defined similarly. We now define the active set A_t as the set of images at iteration t that are in at least one of the positive sets and not in the negative sets

$$A_t = \bigcup_{i=1}^{n_t^+} A_{ti}^+ \setminus \bigcup_{i=1}^{n_t^-} A_{ti}^- \ . \tag{3}$$

Constructing the Most Informative Image Set. To determine the most informative images, we first calculate the information score T_I of each of the active images a at iteration t as the minimum distance to their associated feedback examples

$$T_I(a) = \min_{(s,r)\in S_t^+} d_w\,(s,a)\ . \tag{4}$$

Note that an active image can be in the exploration range of several feedback images. Next, we pick the images with the highest information scores, thus maximizing the minimum distances. As a result, images on the border of our search space will obtain the highest information score.

Constructing the Best Image Set. Besides the most informative images, which allow the user to continue exploring the feature space, we keep an image set that contains the best images thus far. For each active image a at iteration t we calculate a relevance score and those with the highest scores are considered best. A simple way to calculate the relevance score T_R of an active image would be to count how many positive feedback images include this particular image in their exploration range

$$T_R(a) = \sum_{i=1}^{n_t^+} 1_{\{x|d_w(s_i^+,x)<r_i^+\}}(a)\ , \tag{5}$$

where $\mathbf{1}_A(x)$ is an indicator function, indicating the membership of x in set A. An alternative way is to give each active image a score that is dependent on the distance to its feedback point(s)

$$T_R(a) = \sum_{i=1}^{n_t^+} \frac{1_{\{x|d_w(s_i^+,x)<r_i^+\}}(a)}{(1+\gamma d_w(s_i^+,a))}\ , \tag{6}$$

where $\mathbf{1}_A(x)$ is the indicator function as used before and γ a constant that quantifies the rate of relevance decrease as an active image approaches the border of the exploration range.

Our proposed approach is not only suitable for exploring a single local area in feature space, cf. a single query point, but by adjusting the exploration range other areas can also be reached. For instance, given a certain positive feedback image, a method to discover other relevant images located in a different area in feature space would be the following. First, the exploration range of the positive image is expanded to such an extent that the border is located within the other relevant area in feature space. Second, as the border images obtain the highest information score and thus will be presented to the user, the relevant ones can be explored. Third, the exploration range of the initial positive feedback image can be changed back to what is was before, only holding its relevant nearest neighbors.

4 Experiments

The test database was composed of 3000 images taken from the Ponce Texture Database [10] as shown in Figure 3. The 3000 image test set included the 1000 original textures, and 2000 images which were either randomly rotated or scaled from the original versions by up to 15%, resulting in a set of textures that vary in 3D perspective, shape and orientation. The images in the Ponce database are categorized into 25 classes, where each class thus contains 40 original textures, 40 rotated ones and 40 scaled ones. The images are represented by the MPEG-7 Homogeneous Texture Descriptor [11] and by a grayscale histogram, based on a uniform quantization in 16 bins.

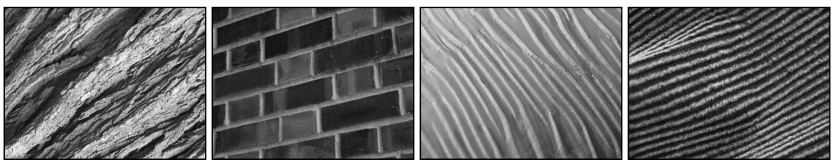

Figure. 3. Example images from the Ponce Texture Database.

For our experiments, we have simulated users that search for images belonging to a certain texture class. We have performed the experiments on two systems, one using our proposed exploration interface and technique ('Explore') and one using a standard interface and the query point movement technique as

proposed by Rocchio ('Rocchio'). As the main strength of our approach is our interface that provides easy access to additional relevant and non-relevant images, we acknowledge that the systems cannot be fairly compared. However, as the systems try to achieve the same goal and are given the same data to work with, we believe that in this sense the systems are comparable. To keep the comparison as fair as possible in all other regards, we have kept the other properties of both interfaces the same, e.g. the number of images on which feedback can be given per iteration.

We have set up a set of experiments for each of the 25 texture classes, where the goal is to find all images belonging to that class within at most 20 iterations. Due to the fact that for each experiment we fill the initial screen with random images, we are affected by the page zero [12] problem. The page zero problem refers to the fact that the retrieval performance depends in great part on which images appear within this initial screen, and specifically how many of these random images belong to the class of interest. Therefore we perform the experiment for each class 100 times and average the results. If the initial screen does not contain any relevant image, we generate a new set of random images until at least one relevant image is shown.

Every iteration the user is presented with 40 images. For the Explore system, these images are composed of the most informative images as calculated by (4). For the Rocchio system, these images consist of the resulting images after performing query point movement. As real users generally don't want to give much feedback, in our simulation per iteration a maximum of 5 images are marked as relevant and a maximum of 5 as non-relevant. Besides the images on which feedback is given, a separate result set is kept that contains the best ranking images. For the Explore system, these images are composed of the top images as calculated by (6). For the Rocchio system, this set is the same set as the feedback set, containing the resulting images after query point movement.

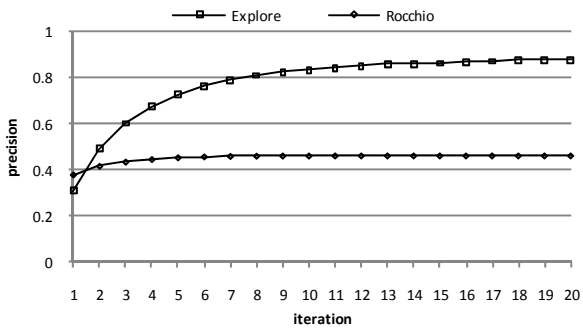

Figure 4. Average precision results.

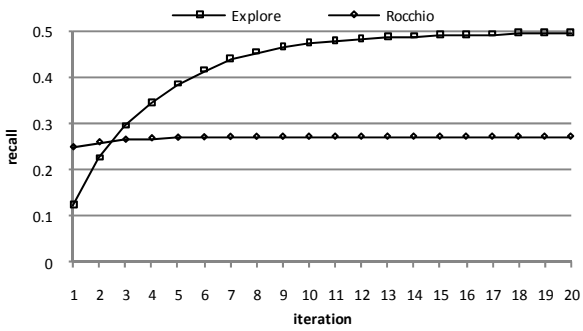

Figure 5. Average recall results.

The results are shown in Figures 4 and 5. For clarity, in our results we define *precision* as the number of relevant images found over the total number of images looked at, which is the top 40 best ranking images, and *recall* as the number of relevant images found thus far over the total number of existing relevant images, which is 120 per texture class. For these experiments, the Explore system does not utilize the aforementioned technique of reaching other areas in feature space, as that would give it an unfair advantage over the Rocchio system.

As we can see in Figure 4, the average precision of relevant images within the 40 best ranking images rapidly increases for the Explore system, reaching 80% after 8 iterations and still improves during following iterations. The Rocchio system finds almost all the relevant images it is able find after the first iteration, and hardly improves after that, reaching a maximum of near 50%. In Figure 5 we can see that the Explore system manages to discover 50% of all relevant images per class, whereas Rocchio finds little over 25%. On average, the accuracy of the Explore system improves considerably over the Rocchio system after only two iterations.

5 Conclusions

In this paper, we have proposed a novel interactive technique that allows the user to visually explore the feature space around relevant images and to focus the search on only those regions in feature space that are relevant. We performed user experiments on a well-known texture database and the results indicate that the new approach leads to an improvement of the amount of relevant images collected. In the future we will focus on adaptive distance measures and feature selection techniques to more optimally explore the feature space around feedback images.

Acknowledgements

Leiden University and NWO BSIK/BRICKS supported this research under grant #642.066.603.

References

[1] Rocchio, J.J. Relevance Feedback in Information Retrieval. In *The Smart Retrieval System: Experiments in Automatic Document Processing*, G. Salton, Ed. Prentice Hall, Englewoods Cliffs, pp. 313-323, 1971.

[2] He, X., Cai, D., Han, J. Learning a Maximum Margin Subspace for Image Retrieval. In *IEEE Transactions on Knowledge and Engineering 20(2)*, pp. 189-201, 2008.

[3] Yu, J., Tian, Q. Learning Image Manifolds by Semantic Subspace Projection. In *ACM International Conference on Multimedia*, pp. 297-306, 2006.

[4] Fan, J., Gao, Y., Luo, H., Jain, R. Mining Multilevel Image Semantics via Hierarchical Classification. In *IEEE Transactions on Multimedia 10(2)*, pp. 167-181, 2008.

[5] Jiang, W., Chan, K.L, Li, M., Zhang, H.J. Mapping Low-Level Features to High-Level Semantic Concepts in Region-Based Image Retrieval. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 244-249, 2005.

[6] Hoi, S.C.H., Lyu, M.R., Jin, R. A Unified Log-Based Relevance Feedback Scheme for Image Retrieval. In *IEEE Transactions on Knowledge and Data Engineering 18(4)*, pp. 509-524, 2006.

[7] Yin, P.-Y., Bhanu, B., Chang, K.-C., Dong, A. Integrating Relevance Feedback Techniques for Image Retrieval using Reinforcement Learning. In *IEEE Transactions on Pattern Analysis and Machine Intelligence 27(10)*, pp. 1536-1551, 2005.

[8] Ortega-Binderberger, M., Mehrotra, S. Relevance Feedback Techniques in the MARS Image Retrieval System. In *ACM Multimedia Systems 9(6)*, pp. 535-547, 2004.

[9] Jin, X., French, J.C. Improving Image Retrieval Effectiveness via Multiple Queries. In *ACM Workshop on Multimedia Databases*, pp. 86-94, 2003.

[10] Lazebnik, S., Schmid, C., Ponce, J. A Sparse Texture Representation using Local Affine Regions. In *IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8)*, pp. 1265-1278, 2005.

[11] Ro, Y.M., Kim, M., Kang, H.K., Manjunath, B.S., Kim, J. MPEG-7 Homogeneous Texture Descriptor. In *ETRI Journal 23(23)*, pp. 41-51, 2001.

[12] La Cascia, M., Sethi, S., Sclaroff, S. Combining Textual and Visual Cues for Content-based Image Retrieval on the World Wide Web. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 24-28, 1998.

[13] Chen, Y.-S., Shahabi, C. Yoda, An Adaptive Soft Classification Model: Content-Based Similarity Queries and Beyond. In *ACM Multimedia Systems 8(6)*, pp. 523-535, 2003.

[14] Datta, R., Joshi, D., Li, J., Wang, J.Z. Image Retrieval: Ideas, Influences, and Trends of the New Age. In *ACM Computing Surveys 40(2)*, article 5, 2008.

[15] Nguyen, G.P., Worring, M. Similarity Learning via Dissimilarity Space in CBIR. In *ACM Workshop on Multimedia Information Retrieval*, pp. 107-116, 2006.

[16] Zavesky, E., Chang, S.-F., Yang, C.-C. Visual Islands: Intuitive Browsing of Visual Search Results. In *ACM Conference on Image and Video Retrieval*, pp. 617-626, 2008.

[17] Jain, R. Experiential Computing. In Communications of the ACM 46(7), pp. 48-55, 2003.

[18] Lew, M.S., Sebe, N., Djeraba, C., Jain, R. Content-Based Multimedia Information Retrieval: State of the Art and Challenges. In *ACM Multimedia Systems 2(1)*, pp. 1-19, 2006.