Dynamic Queries with Relevance Feedback for Content Based Image Retrieval

Murat Birinci*, Esin Guldogan*, and Moncef Gabbouj

Department of Signal Processing, Tampere University of Technology, Tampere, Finland
{murat.birinci,esin.guldogan,moncef.gabbouj}@tut.fi

Abstract. A novel relevance feedback scheme utilizing dynamic queries for content based image retrieval systems is proposed, where the retrieval results are updated instantly based on the user's feedback. The user is expected to label at least one image as positive or negative, revealing the gist of the expected retrieval results. Then the retrieval results are updated dynamically, without any further user interaction, based on the similarity of the query and the selected image in different feature spaces increasing the semantic accuracy of the retrieval. The proposed method not only invalidates the drawbacks of current relevance feedback systems in terms of user experience, but also provides an innovative stand point for the relevance feedback scheme as well.

Keywords: dynamic queries, relevance feedback, content based image retrieval.

1 Introduction

Developments in multimedia technology in the past century altered the media world from analog to digital, resulting in not only putting every individual into a content creator position, but also transferring all the media already at hand into digital format, which significantly increases the amount of digital media available. Visual, aural and textual information can now be created, stored and modified digitally via everyday devices or software such as cameras, audio players, text editors, implying every single user may have a vast amount of such media. Moreover, improvements in network technology, especially in Internet, give rise to distribution of this content over an indefinitely large population, yet bringing forth the problem of management of this information for efficient accessibility. The very first approaches on multimedia indexing utilize text-based annotations, analogous to library catalogues, where a text based description is linked to every database element. However the reliability of such methods is entirely annotator dependent and furthermore, they require vast amount of work especially with increasing database sizes. Such drawbacks are also encountered for any kind of manual indexing method which is why automatic indexing is essential. Automatic schemes intend to overcome any subjectivity and represent the content effectively, defining the scope of *Content Based Multimedia Retrieval* (CBMR).

* These authors contributed equally to this work.

J.A. Jacko (Ed.): Human-Computer Interaction, Part I, HCII 2011, LNCS 6761, pp. 547–554, 2011.

Without any loss of generalization, in this paper we reduce our focus to *Content Based Image Retrieval* (CBIR) which deals with visual features only. Various visual features, such as color, texture and shape, are utilized in CBIR systems in order to describe the content of an image (or any visual media); however their retrieval performance is usually limited especially on large databases due to the lack of discrimination power of such features. Moreover, a more evident reason is while such features strive to extract an objective description of the content, the content is undoubtedly subjective and there is the so called *semantic gap* between the description of the content and its semantic interpretation at the user side. Therefore, incorporation of user subjectivity and experience is of decisive importance in order to achieve competent retrieval results.

Relevance feedback is found to be one of the most powerful methods for improving image retrieval performance and bridging the semantic gap [1]. The word *relevance* in this context is described as "*the ability of a retrieval system to retrieve material that satisfies the needs of the user*". Relevance feedback methods predict and learn user's preference to improve retrieval results. They interact with user during the query in order to get the user's subjective perception for improving the query results via providing the opportunity for evaluating them. It iteratively improves the accuracy of the retrieval by modifying the query, based on the user's feedback. The iterated relevance feedback may guarantee improved retrieval results. However, while being such a supportive and practical tool, most of the relevance feedback methods suffer from user experience side due to its iterative nature. Users are expected to update their queries after each feedback in order the system to learn their behavior and choices. Thus, it is obvious that current CBIR systems are lacking the adjustment to average users' needs, i.e. they cannot synchronize to the way an average user searches, even though the technology is available. Zhang accurately stated the same issue in [2] as "*the user interface design should adapt to users' behavior, not shape users' behavior*". Yet, CBIR society mainly focused on the background technology instead of how to utilize that technology using an appropriate design in order to reach the average user.

The rest of the paper is organized as follows. Section 2 provides an overview of the related work in this area. In section 3, the proposed method is explained in detail and experimental results are given in section 4. Section 5 concludes the paper and suggests possible future work.

2 Related Work

There are various proposed relevance feedback methods in the literature. Most of them treat each iteration as a separate query, i.e. by combining the initial query and the feedback from the user they form a new query and present the reconstructed results [1]. Various feature re-weighting and query re-formulating methods in order to form the new query are introduced in the literature [3], [4]. For example Djordjevic and Izquierdo in [5], used variance in order to describe discrimination power of the feature. Main drawback of such approaches is the necessity of large amount labeled data and several iterations for improving the semantic accuracy. More recent approaches involve neural networks and machine learning approaches [6], [7], in order to learn and model user behavior. However, the computational complexity significantly increases when the learning process is involved. Moreover, it should be noted that the user's preferences might vary in every query. For instance, the user might use the picture in Fig. 1 in order to search for *horses*. However the same picture can be used for searching *beach* or *ocean* another time. Therefore, if the system *learns* that this picture is associated with horses, future searches with this picture will yield results with more horses in them. This is a clear example of what is meant in Section 1 as "shaping users' behavior". However, users' interests and objectives are not the same at all times.

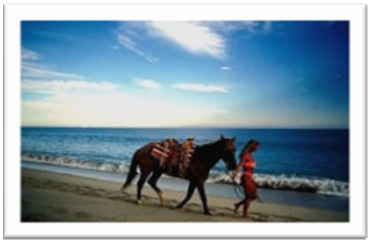


Fig. 1. A picture with various content

An instance-based relevance feedback method is proposed by Giacinto *et al.* [8], where they represent the images in a dissimilarity-space. Their notion of relevance is the degree of similarity of an image to the nearest relevant (based on user feedback) image. Similarly, irrelevance is the dissimilarity from the nearest irrelevant image. Than the images are ranked again according to their relevance score and top K images are displayed. However, while the method points out the significance of query-based feedback instead of an overall learning approach, re-calculation of distances and updating the whole query is no different that re-making the query for the user. Moreover, the number of images expected from the user to mark as relevant or irrelevant is in the order of dozens. Those are again facts that are "shaping users' behavior" instead of "adapting it".

Another instance-based method is proposed in [9], where user feedback is used in re-weighting individual feature distances based on intra- and inter-cluster relations. While this method avoids re-calculation of feature distances (which is computationally expensive and time consuming) and simply re-weights the already calculated distances, it requires an additional query after the feedback using the re-weighted distances. This means re-ranking of *all* images in the database.

Shneiderman presented "direct manipulation" techniques in 1983 [10] and "dynamic queries" in 1994 [11] in order to introduce quick and powerful query methods for database and information retrieval via graphical controls such as sliders, buttons etc. Dynamic queries describe the interactive user control of visual query parameters that generate a rapid animated visual display of database search results. They enable users to conveniently overview, explore and filter out the results and observe the effects immediately. Therefore, the user can obtain the desired results in real-time without any reformulation of the query. In this context, utilization of dynamic queries in relevance feedback methods should be much more than a mere process of gathering relevance information. The feedback should provide a user preference and influence the results instantly.

3 Proposed Method

We propose an instance-based relevance feedback method that utilizes small number of labeled data at that particular query session, i.e. the relevance feedback received from the user affects only the current query and not the future retrievals. The proposed scheme utilizes dynamic querying meaning that the user is not required to repeat or reformulate the query. Instead, the results are updated dynamically based on the feedback.

Users are expected to label the image(s) either positively or negatively according to their subjective preferences. For positively labeled image(s), it is assumed that user wants more images similar to that particular image(s), thus that image(s) will be used for presenting new images to the user dynamically. Similarly, users may negatively label the image(s) for their subjective preferences, meaning that such images are out of user's interest. Based on the negatively labeled images, particular images will be eliminated and new images will be introduced to the user by modeling the user's preferences. Fig. 2 demonstrates how a negatively labeled image updates the current query. Initial retrieval results based on text based search are shown in Fig. 2.a (Yet, we used only visual features in our experiments). User is provided to choose any image as preferred or non-preferred when hovered. In Fig. 2.b, the user does not prefer to have the image in his results and the system removes that image *together with the ones having similar content* (Fig. 2.c) and brings new images from lower ranks not including similar ones to the user's choice (Fig. 2.d).

Fig. 2. An illustration of how negative feedback affects the results

The overall similarity distance between two images is calculated by averaging similarity distances from several different features. If N distinct features are extracted from images I_1 and I_2, than the overall similarity distance D between those two images is calculated as:

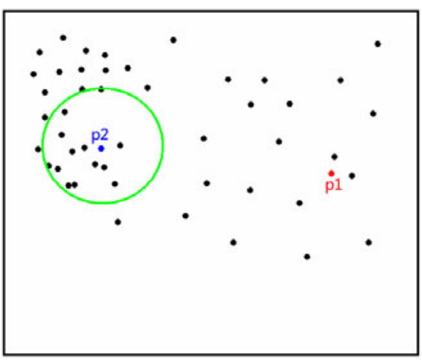
$$D(I_1, I_2) = \frac{1}{N}\sum_{i=1}^{N} d_i(I_1, I_2) \tag{1}$$

where $d_i(I_1,I_2)$ is the similarity distance between I_1 and I_2 for the i^{th} feature. However, the fact that relevance feedback is necessary means that the user is not satisfied with the discrimination provided by D. Hence, the user provides his/her preference on which images should be more similar (positively labeled images) and which should be regarded as different (negatively labeled images).

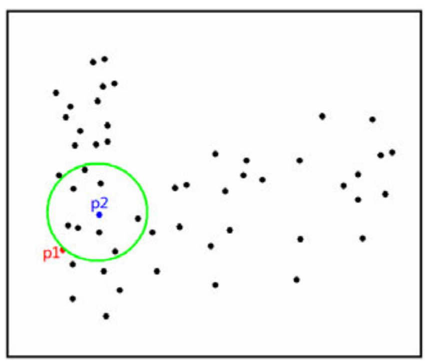
It should be noted that all the labeled images are found to be similar based on the overall similarity distance. Thus, we analyze the feature spaces independently and seek a feature space that provides the requested (dis-)similarity.

Let us consider the positive and negative feedback cases separately. If the user labels an image (namely I_2) as "irrelevant", a feature providing a high similarity distance from the query image (namely I_1) is sought. Therefore we select the i^{th} feature space that provides the maximum similarity distance, i.e. $max(d_i(I_1,I_2))$. In order to remove the images that are similar to I_2 and dis-similar to I_1, we remove the images that have small similarity distance to I_2 in the i^{th} feature space (Fig. 3). The decision on the number of images to be removed can be made in many ways. A distance threshold can be set and images closer than this threshold to I_2 can be removed. This is also illustrated in Fig. 3 where the distance threshold is shown as the green circle around *p2*. In our experiments we used *KNN* (K-Nearest-Neighbors) algorithm in order to remove K images nearest to I_2 in the i^{th} feature space.

On the other hand, if an image is labeled as "relevant", we find the i^{th} feature space yielding the minimum similarity distance with the query image and i.e. $min(d_i(I_1,I_2))$. Similarly, the K nearest images to I_2 are found and raised to higher ranks. However, while it was easy to simply discard K irrelevant images, more analysis is needed in



(a)



(b)

Fig. 3. Two feature spaces that provide different similarity distances between the same elements p1 and p2

order to determine the updated ranks of the *K* relevant images. Incautiously moving the *K* relevant images to the top *K* rank would be interfering with the users' preferences. Thus, in order to include those *K* images among the presented images to the user and determine their appropriate ranks, one should consider both their distances in the i^{th} feature space and the distances of the presented images. Therefore, the distances of *K* relevant images are than re-weighted according to the following formulas:

$$\mu_i = \frac{1}{K}\sum_{m=1}^{K} d_i(I_1, I_m), \tag{2}$$

$$\mu_\forall = \frac{1}{L}\sum_{m=1}^{L} D(I_1, I_m), \tag{3}$$

$$D'(I_1, I_m) = \frac{\mu_i}{\mu_\forall} d_i(I_1, I_m) \tag{4}$$

where *L* is the number of images presented to the user. In other words, in order to map the distances of *K* relevant images to the *L* presented images, their distances are weighted with the ratio of average distances (μi and $\mu_\forall$). The value of *L*, i.e. the number of images presented to the user, depends on the retrieval system and the UI. Hence, in order to promote the *K* relevant images among the presented images, the average similarity distance of the *L* presented images is used to weight the distances of *K* relevant images.

4 Experimental Results

In our experiments, we used Corel real-world image databases for evaluating the retrieval results of the proposed method. Corel image data sets are well-categorized and widely used in the CBIR literature. For evaluating the results, a Corel database with 1000 images are used. These images are pre-assigned by a group of human observers to 10 semantic classes each containing 100 images. The classes are: Africa, Beach, Buildings, Buses, Dinosaurs, Flowers, Elephants, Horses, Food, and Mountains. In our experiments, the following low-level color, shape, and texture features are used: YUV and HSV color histograms with 128-bins, Gray Level Co-Occurrence Matrix texture feature with parameter value 12, Canny Edge Histogram, and Dominant Color with 3 colors. We compared our results with the methods in [8] and [5] which are briefly explained in Section 2. Average precision values are calculated based on the retrieval results from these queries. We used *K*=10 for the calculation of *KNN* as described in Section 3. For our system the number of images presented to the user is 24, i.e. *L*=24.

We performed 17 queries from different classes. Fig. 4 shows the average precision values calculated for the initial query (no user feedback), the proposed method and the competing methods in [8] and [5]. It is clear that the improvement in precision using the proposed method is slightly better than the competing methods. While the performance measures are close to each other, another point that should be considered is the amount of user feedback. Users are not usually in favor of laborious work such as providing feedback. As we discussed in Section 2, the system should not require immoderate user effort, i.e. it should not shape users' actions. While the methods in [8] and [5] require significant amount of feedback (labeled images), the proposed method is designed to work with minimum amount of labeled data (images). Therefore we used 1 positively labeled (relevant) and 1 negatively labeled (irrelevant) image as the user feedback for the proposed method. However, since the competing methods are not designed to work with such limited input, we used 3 positive and 3 negative feedbacks in order to obtain the results in Fig. 4. Even with such a difference in the user input, the proposed method proves to perform better than the competing techniques. Moreover, the computational efficiency of the proposed method outperforms the competing methods. Since other methods require complex computations including calculation of variances etc. our method works 7 times faster than [8] and 50 times faster than [5]. This fact also enables us to utilize dynamic querying by working under the limitations stated in [11].

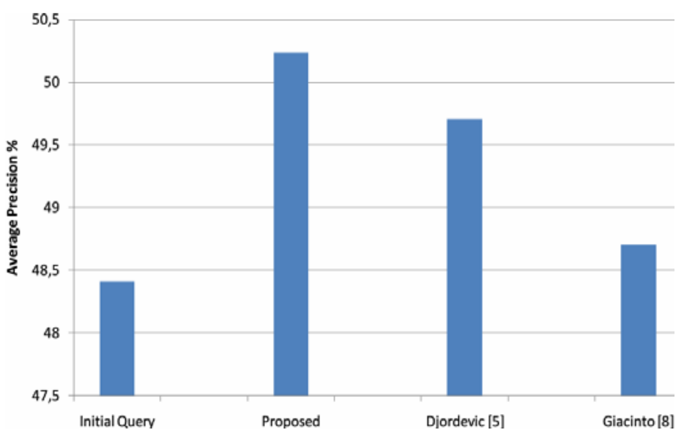


Fig. 4. Performance results for the initial query, the proposed and competing methods

5 Conclusion

We proposed a novel relevance feedback method for content based image retrieval. Dynamic querying techniques are used in order to instantly update the retrieval results. The proposed method treats each query independently, enabling users to have different preferences every time they query. Minimum amount of feedback is required in order to grasp user preference. User feedback is utilized in order to find the feature space that reflects user's view of similarity between the query image and the selected image. Experimental results prove that the proposed method outperforms competing methods both in performance and computational efficiency. The computational efficiency together with dynamic querying and the limited amount of required user labor provides a better user experience than the compared to the state-of-art methods.

References

1. Zhou, X., Huang, T.S.: Relevance Feedback for Image Retrieval: A Comprehensive Review. ACM Multimedia Systems 8, 536–540 (2003)
2. Zhang, D.: A Survey of User Interfaces in Content-based Image Search Engines on the Web. In: Proceedings of the Fourth Workshop on Human-Computer Interaction and Information Retrieval (HCIR), pp. 129–132 (2010)
3. Heesch, D., Rueger, S.: Two step relevance feedback for semantic disambiguation in image retrieval. In: Sebillo, M., Vitiello, G., Schaefer, G. (eds.) VISUAL 2008. LNCS, vol. 5188, pp. 204–215. Springer, Heidelberg (2008)
4. Piras, L., Giacinto, G.: Neighborhood-Based Feature Weighting for Relevance Feedback in Content Based Retrieval. In: Workshop on Image Analysis for Multimedia Interactive Services, London, UK (2009)
5. Djordjevic, D., Izquierdo, E.: An Object- and User-Driven System for Semantic-Based Image Annotation and Retrieval. IEEE Trans. Circuits Syst. Video Techn. 17(3), 313–323 (2007)
6. Rao, Y., Mundur, P., Yesha, Y.: Fuzzy SVM Ensembles for Relevance Feedback in Image Retrieval. In: Proc. of CIVR, pp. 350–359 (2006)
7. Zhang, L., Lin, F., Zhang, B.: Support Vector Machine Learning for Image Retrieval. In: Proc. of ICIP, pp. 721–724 (2001)
8. Giacinto, G., Roli, F.: Instance-based relevance feedback for image retrieval. In: Perner, P. (ed.) Case-Based Reasoning on Images and Signals, pp. 489–496. Springer, Heidelberg (2008)
9. Guldogan, E., Gabbouj, M.: Dynamic Feature Weights with Relevance Feedback in Content-Based Image Retrieval. In: Proceedings of the IEEE International Symposium on Computer and Information Sciences, ISCIS, Cyprus (2009)
10. Shneiderman, B.: Direct Manipulation: A Step Beyond Programming Languages. Computer 16, 57–69 (1983)
11. Shneiderman, B.: Dynamic queries for visual information seeking. IEEE Software 11, 70–77 (1994)