Strategies for positive and negative relevance feedback in image retrieval*

Henning Müller, Wolfgang Müller, Stéphane Marchand-Maillet and Thierry Pun
Computer Vision Group, University of Geneva, CH-1211 Geneva 4, Switzerland
Henning.Mueller@cui.unige.ch tel. +41 22 705 7633, fax. +41 22 705 7780

David McG. Squire
School of Computer Science and Software Engineering, Monash University, Melbourne, Australia

Abstract

Relevance feedback has been shown to be a very effective tool for enhancing retrieval results in text retrieval. In content-based image retrieval it is more and more frequently used and very good results have been obtained. However, too much negative feedback may destroy a query as good features get negative weightings.

This paper compares a variety of strategies for positive and negative feedback. The performance evaluation of feedback algorithms is a hard problem. To solve this, we obtain judgments from several users and employ an automated feedback scheme. We can then evaluate different techniques using the same judgments. Using automated feedback, the ability of a system to adapt to the user's needs can be measured very effectively. Our study highlights the utility of negative feedback, especially over several feedback steps.

1. Introduction

Relevance feedback (RF) has shown to be extremely useful in text retrieval (TR) applications [7], and is now being applied in some content-based image retrieval systems (CBIRSs) [5, 9]. Since human perception of image similarity is both subjective and task-dependent [10, 1], we believe RF to be an essential component of a CBIRS. By augmenting the query with features from relevant and non-relevant retrieved images, a query can be produced which better represents the user's information need.

Performance evaluation is a difficult problem in content-based image retrieval, largely due to the subjectivity and task-dependence issues mentioned above. For these reasons evaluation *must* involve experiments with *several* real users. Examples of such studies exist but much published work contains little or no quantitative performance evaluation. The CBIR community still lacks a commonly accepted database of images, queries and relevance judgments, such as the TREC databases used in TR.

The evaluation of retrieval performance has been thoroughly studied in the TR community [6]. One of the most common measures, the *Precision* vs. *Recall* (PR) graph [6, 11], is now increasingly used in CBIR [8, 9]. In this paper, performance results are presented in the form of PR-graphs averaged over several users and several queries.

To evaluate the interactive performance of a system and the effectiveness of RF, new measures need to developed. These can be based on relevance judgments by real users and automated feedback to evaluate the ability of a system to adapt to the user's needs.

2. Related work

In TR, RF was introduced as early as the late 60's (*e.g.* in the SMART system), and was shown to improve results significantly. It was shown later that the use of negative feedback could enhance performance strongly. However, too much negative feedback can "destroy" a query. Consequently, it was proposed that the positive and negative components be weighted separately [4] (see §4.1.4).

The use of RF in CBIR is more recent, and fewer feedback strategies have been investigated, especially for negative feedback. Huang and Mehrotra propose several levels of feedback and get better results than before feedback [5]. In `PicHunter`, Bayesian feedback is used to present the user with choices which maximise information gain when searching for a given target. It is often stated that the systems perform better after feedback, but quantitative measurements are seldom done.

3. The *Viper* system

The *Viper* system, inspired by TR systems, uses a very large number of simple features[1]. The present version em-

*This work was supported by the Swiss National Foundation for Scientific Research (grant no. 2000-052426.97).

[1] Visual Information Processing for Enhanced Retrieval. Web page: `http://cuiwww.unige.ch/~viper/`

ploys both local and global image color and spatial frequency features, extracted at several scales, and their frequency statistics in both the image and the whole collection. The intention is to make available to the system low-level features which correspond (roughly) to those present in the human vision system.

More than 80000 features are available to the system. Each image has $\mathcal{O}(10^3)$ such features, the mapping from features to images being stored in an inverted file. The use of such a data structure, in conjunction with the feature weighting scheme, means that textual features are treated in exactly the same way as visual ones. Further details about the architecture of the *Viper* system can be found in [9].

We use 2500 diverse images supplied by Télévision Suisse Romande. In the experiment, 3 users gave judgments for 14 query images. The users chose different and varying numbers of relevant images for each query. These experiments are described in detail in [3].

4. Feedback strategies

The two main strategies for RF are either (1) to make separate queries for each feedback image and merge the query results or (2) to create a "pseudo-image" from the feedback images and execute a query with this image. *Viper* uses the second method by combining the features from the feedback images and normalizing their frequencies.

4.1. Automated feedback

Automated RF can be applied once user judgments for an image collection exist. Thus a reproducible RF for every user can be simulated based upon the judgments and the initial query results of a system. Via this technique, the flexibility of a system with respect to users' needs can be measured, *e.g.* by feeding back the images the user judged as relevant and which were returned in the top $n = 20$ of a query result. This technique can be used to compare different feedback strategies or to enhance user queries by automatically creating negative feedback.

4.1.1. Only positive feedback

Positive feedback is limited to preselected images and weights the features of these images more strongly. As all high ranked returned images have many features in common, the non-relevant images may also be ranked highly in the next step. For this feedback, we select as relevant all the images from the initial query result which the user judged to be relevant. We chose images for feedback from the first 20 highest ranked response images, which is a reasonable number to display on screen simultaneously. 50 is regarded as the maximum number of images a user might normally browse, and 100 is used to show the improvements.

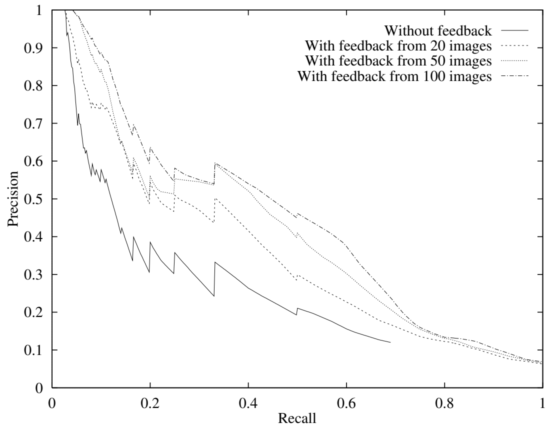

Figure 1. Effect of positive feedback.

The improvement in performance using RF is quite large as can be seen in Figure 1. When using only feedback from the first 20 result images, the PR-graph is improved by 20% in some areas. Using 50 images for RF gives an additional improvement of about 10% in most regions. The use of 100 images improves only some parts of the graph by an additional 5%. Some of the improvement comes only from relevant images being ranked higher in the top n and not from returning new relevant images.

4.1.2. Positive and negative feedback

Negative feedback can improve the query result greatly, but it is important to use the right images as negative feedback so as not to inhibit any important features. Many systems have problems with too much negative feedback. Based on these facts, we apply a variety of methods for automatic selection of negative RF. Positive images from the top 20 returned were still all selected as positive feedback. As negative feedback, we chose the first two and the last two non-relevant answer images. Since they influence different parts of the PR-graph we also combine the two strategies.

We can see in Figure 2 that returning the first two images as negative feedback improves the beginning of the PR-graph by 4 to 5%; using the last two improves the middle of the PR-graph by up to 7%. The combination of both improves all parts of the graph by up to 9%. This shows that different negative feedback images improve different parts of the graph significantly by removing different areas of feature space from the query.

With this knowledge, a query from a user who only uses positive feedback can be improved by automatically supplying non-selected images as negative feedback.

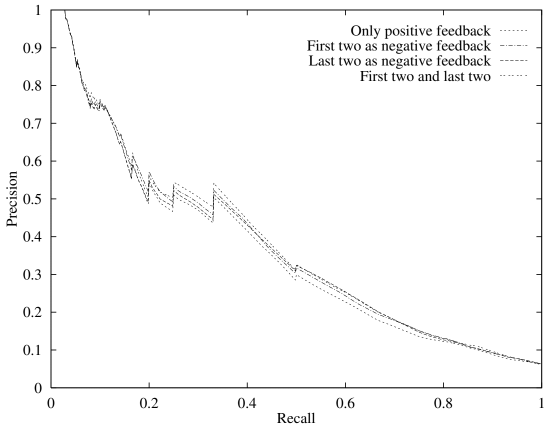

Figure 2. Different negative feedback images.

4.1.3. Different feedback weightings

As we know that different negative feedback images can improve different parts of the PR-graph but also decrease performance when used in excess. We minimize the latter effect by weighting the images with a factor other than -1, we can feed back all neutral images as negative RF.

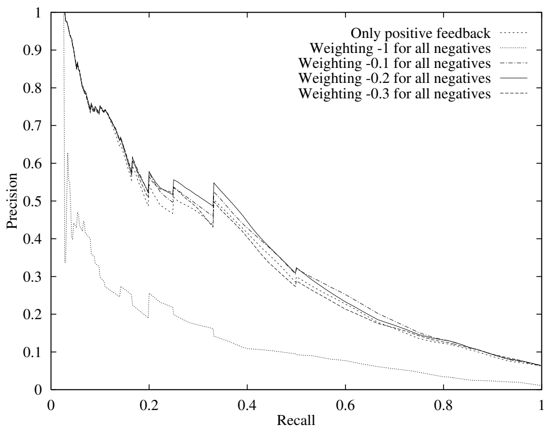

Figure 3. Various feedback weightings.

In Figure 3, we can see that the value of -0.2 yields the best curve in most areas, only in the end the curve with -0.3 is better but these last parts of a PR-graph are not as important since they only give information about images which are not shown to the user. The -0.3 curve is sometimes even worse than the curve with only positive feedback. The value of -0.2 creates improvements of up to 7 or 8%. Using higher weightings does not bring any further improvements.

A good idea might be to create negative feedback automatically with a low weighting when the user does not use any or enough negative feedback.

4.1.4. Separately weighted feedback

Problems due to too much negative feedback in TR were addressed by Rocchio in the 60s [4]. Following this work, our system weights the features of positive and negative query images separately according to Equation 1,

$$Q = \frac{\alpha}{n_1} \sum_{i=1}^{n_1} R_i - \frac{\beta}{n_2} \sum_{i=1}^{n_2} S_i, \qquad (1)$$

where Q is the set of weighted features making up the query, n_1 and n_2 are the numbers of positive and negative images in the respectively, R_i and S_i are the (possibly weighted) features in the postive and negative images, and α and β determine the relative weightings of the positive and negative components of the query. We use $\alpha = 0.65$ and $\beta = 0.35$.

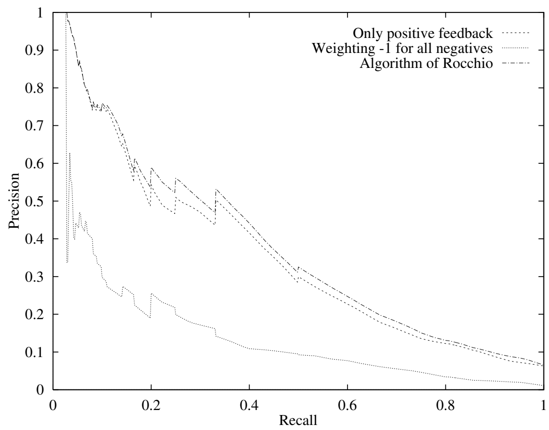

Figure 4. RF with modified Rocchio algorithm.

This technique significantly improves the query results (up to 9%). This is better than the other methods for positive and negative feedback. Clearly, we still need to test whether the weightings of 0.65 and 0.35 are as good for CBIR as they proved to be for TR, but we already made the result more or less independent from the number of positive and negative feedback images. Using this method with a larger number of result images (*e.g.* 50 as in §4.1.1) improves the results even more.

4.1.5. Several steps of feedback

To measure the interactive performance of a system, we need to consider more than one step of RF since browsing is a crucial task for CBIR [2]. We thus made experiments with several steps of RF.

Figure 5 shows the results using two steps of only positive feedback. The major improvement occurs at the first feedback step (20%). For the second step, it is rather small (2 to 3%). The improvement with positive and negative

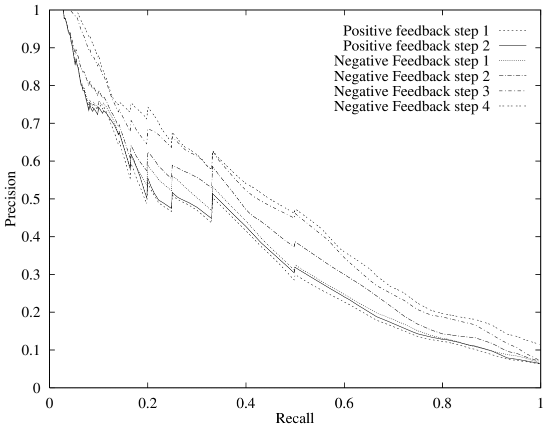

Figure 5. Several feedback steps.

feedback is remarkable for the first four steps where the results continuously get better. The first step already shows an improvement of about 25% and the second step an additional 10%. In the third step the result improves by about 10% in the beginning and by 8% in the middle parts. The gain for the fourth is 5% in the middle and as well in the end. This improvement in the end means that images which were far away from the initial query have been moved closer.

These results show the great importance of negative RF for the browsing process. The effect of positive feedback almost disappears after only one or two steps so the possibility to move in feature space is limited. Negative feedback offers many more options to move in feature space and find target images. Even hard queries are continuously improved at each feedback step. This flexibility to navigate in feature space is perhaps the most important aspect of a CBIRS.

5. Conclusions

In this article we show the influence of various RF strategies on the query result. RF always improves the results. However, too much negative feedback can destroy the query. This can be avoided by using Rocchio's technique of separately weighting positive and negative features. We showed that several steps of positive and negative feedback increasingly enhance the query results, thus allowing navigation within the database. Using a larger number of images as a source for feedback improves results, but this potential is limited by the number of images a user really inspects.

Using a variety of automated RF strategies, we can evaluate the flexibility of a CBIRS. It is important that using several steps of feedback continues to improve the results so, that feature space can be explored thoroughly. Several steps of positive and negative RF can form a basis for evaluating the interactive performance of a CBIRS.

The good performance of negative RF leads to the idea of automatically feeding back neutral images as negative if none are provided by the user. This can help novice users to get better results.

References

[1] Y. H. Kim, K. E. Lee, K. S. Choi, J. H. Yoo, P. K. Rhee, and Y. C. Park. Personalized image retrieval with user's preference model. In C.-C. J. Kuo, S.-F. Chang, and S. Panchanathan, editors, *Multimedia Storage and Archiving Systems III (VV02)*, volume 3527 of *SPIE Proceedings*, pages 47–55, Boston, Massachusetts, USA, November 1998.

[2] M. Markkula and E. Sormunen. Searching for photos - journalists' practices in pictorial IR. In J. P. Eakins, D. J. Harper, and J. Jose, editors, *The Challenge of Image Retrieval, A Workshop and Symposium on Image Retrieval*, Electronic Workshops in Computing, Newcastle upon Tyne, 5–6 February 1998. The British Computer Society.

[3] H. Müller, D. M. Squire, W. Müller, and T. Pun. Efficient access methods for content-based image retrieval with inverted files. In S. Panchanathan, S.-F. Chang, and C.-C. J. Kuo, editors, *Multimedia Storage and Archiving Systems IV (VV02)*, volume 3846 of *SPIE Proceedings*, Boston, Massachusetts, USA, September 20–22 1999.

[4] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System, Experiments in Automatic Document Processing*, pages 313–323. Prentice Hall, Englewood Cliffs, New Jersey, USA, 1971.

[5] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool in interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):644–655, September 1998. (Special Issue on Segmentation, Description, and Retrieval of Video Content).

[6] G. Salton. The state of retrieval system evaluation. *Information Processing and Management*, 28(4):441–450, 1992.

[7] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–287, 1990.

[8] J. R. Smith and S.-F. Chang. VisualSEEk: a fully automated content-based image query system. In *The Fourth ACM International Multimedia Conference and Exhibition*, Boston, MA, USA, November 1996.

[9] D. M. Squire, W. Müller, H. Müller, and J. Raki. Content-based query of image databases, inspirations from text retrieval: inverted files, frequency-based weights and relevance feedback. In *The 11th Scandinavian Conference on Image Analysis (SCIA'99)*, pages 143–149, Kangerlussuaq, Greenland, June 7–11 1999.

[10] D. M. Squire and T. Pun. A comparison of human and machine assessments of image similarity for the organization of image databases. In M. Frydrych, J. Parkkinen, and A. Visa, editors, *The 10th Scandinavian Conference on Image Analysis (SCIA'97)*, pages 51–58, Lappeenranta, Finland, June 1997.

[11] J. Tague-Sutcliffe. The pragmatics of information retrieval experimentation, revisited. In K. Spark Jones and P. Willett, editors, *Readings in Information Retrieval*, Multimedia Information and Systems, chapter 4, pages 205–216. Morgan Kaufmann, 340 Pine Street, San Francisco, USA, 1997.