Relevance feedback and term weighting schemes for content-based image retrieval

David Squire, Wolfgang Müller, and Henning Müller

Computer Vision Group, Computer Science Department, University of Geneva,
rue Général Dufour, 1211 Geneva 4, Switzerland.
squire@cui.unige.ch, muellerw@cui.unige.ch, muellerh@cui.unige.ch

Abstract. This paper describes the application of techniques derived from text retrieval research to the content-based querying of image databases. Specifically, the use of inverted files, frequency-based weights and relevance feedback are investigated. The use of inverted files allows very large numbers ($\geq \mathcal{O}(10^4)$) of *possible* features to be used. since search is limited to the subspace spanned by the features present in the query image(s). A variety of weighting schemes used in text retrieval are employed, yielding different results. We suggest possibles modifications for their use with image databases. The use of relevance feedback was shown to improve the query results significantly, as measured by precision and recall, for all users.

1 Introduction

In recent years the use of digital image databases has become common, both on the web and for preparing electronic and paper publications. The efficient querying and browsing of large image databases has thus become increasingly important. Content-based retrieval from large text databases has been studied for more than forty years, yet the insights and techniques of text retrieval (TR) have largely been ignored by content-based image retrieval (CBIR) researchers, or reinvented without heeding the prior work. The utility of *Relevance Feedback* (RF) is long-established [1], yet its application in CBIR systems (CBIRSs) is very recent. Similarly, a great variety of term-weighting approaches have been investigated, both empirically and theoretically [2]. Means of system evaluation have also been thoroughly studied [3], yet *Precision* and *Recall*, the usual performance measures, are ignored by many researchers in CBIR.

TR systems usually treat each possible term (*i.e.* word) as a dimension of the search space. Spaces with $\mathcal{O}(10^4)$ dimensions are thus typical. The key realization is that in such systems both queries and stored objects are sparse: they have only a small subset ($\mathcal{O}(10^2)$) of all possible attributes. Search can thus be restricted to the subspace spanned by the query terms. The data structure which makes this efficient is the *Inverted File* (IF), described in §3.2. Conversely, considerable effort has been devoted by CBIR researchers to the search for compact image representations (choosing the "right" features), and to the use of techniques such as factor analysis [4] to reduce the feature space dimensionality.

We present a CBIRS which uses an IF, with more than 80000 possible features per image. Using 10 queries for each of 5 users on a test database of 500 images, we compare the effectiveness of a variety of feature-weighting schemes derived from TR. Modifications to these schemes, specific to CBIR, are suggested. We analyze the performance of these weighting schemes both with and without RF, and that of a typical low-dimensional, nearest-neighbour CBIRS, using precision and recall graphs. The TR-inspired weighting schemes are found to improve performance, and the addition of RF makes a still greater difference.

2 Current CBIR research

CBIR researchers acknowledge that the general computer vision problem remains unsolved: semantic retrieval is impossible. The usual approach is to extract low-level features and an attempt to capture image similarity using some function of them. Object recognition is not attempted. Most systems employ features based on colour, texture or shape. Features are often computed globally, and contain no spatial information. Some systems allow the user to influence the relative weights of these classes of features.

2.1 Features

By far the most commonly used feature is colour (*e.g.* [5–7]), usually computed in a colour space thought to be "perceptually accurate" (*e.g.* HSV [7] or CIE [8]). The usual representation is the colour histogram. *Histogram intersection* is the most frequently used distance measure. A disadvantage is that this takes no account of perceptual similarities between bins. Measures exist which use a matrix of bin similarity coefficients [5], but the choice of coefficients is not obvious, and the cost is quadratic.

Many systems use texture to improve image characterization A great variety of texture features has been employed: hierarchies of Gabor filters [9]; the Wold features used in Photobook [10]; the coarseness, contrast, and directionality features used in QBIC [5]; and many more.

Shape features are often computed assuming that images contains only one shape, and are thus best applied to restricted domains. Shape features include: modal matching, applied to isolated fish, rabbits and machine tools [11]; histograms of edge directions, applied to trademarks [6]; matching of shape components such as corners, line segments or circular arcs [12].

Global features are inadequate for many CBIR tasks: users may be interested in the spatial layout of colours, textures and shapes, or in particular objects. One approach is to use features which retain spatial information, such as wavelet decompositions [13]. Others segment the image into regions, and then extract features such as color and texture from them, as well as spatial properties such as size, location and their relationships to other regions [7, 14, 15]. This turns CBIR into a labeled graph matching problem.

2.2 Similarity

The meaning of similarity in CBIR is rarely addressed, yet it is vital to do so: human judgments of similarity vary greatly [16]. Image similarity is typically defined using a metric on a feature space. It is often implied that if one chooses the "right" features proximity in feature space will correspond to perceptual similarity. There are several reasons to doubt this, the most fundamental being the *metric assumption*. There is evidence that human similarity judgments do not obey the requirements of a metric: "[Self-identity] is somewhat problematic, symmetry is apparently false, and the triangle inequality is hardly compelling" [17, p. 329]. The lack of symmetry is the most important issue: the features which are significant depend on which item is the query.

Some attempts have been made to address these problems. Self-organizing maps have been used to cluster texture features according to class labels provided by users [9]. A set-based technique has been applied to learn groupings of similar images from positive and negative examples provided by users [10]. Distance Learning Networks attempt to learn a mapping from feature space to "perceptual similarity space" using human similarity judgment data [18].

2.3 Relevance feedback

There are two basic approaches to RF. According to the RF, a system can create a composite query from relevant and non-relevant images [19], or it can adjust its similarity metric [8]. Some use the variance of features in the relevant set as a weighting criterion [20].

3 The *Viper* system

Viper[1] employs more than 80000 simple colour and spatial frequency features, both local and global, extracted at several scales. The fundamental difference between traditional computer vision and image database applications is that there is a human "in the loop". RF allows a simple classifier to be learnt "on the fly", corresponding to the user's information need.

3.1 Features

Colour features *Viper* uses a palette of 166 colours, derived by quantizing *HSV* space into 18 hues, 3 saturations, 3 values and 4 grey levels. Two sets of features are extracted from the quantized image. The first is a colour histogram, with empty bins are discarded. The second represents colour layout. Each block in the image (the first being the image itself) is recursively divided into four equal-sized blocks, at four scales. The occurrence of a block with a given mode color is treated as a binary feature. For our 256×256 images there are thus 56440 possible colour block features, of which each image has 340.

[1] `http://cuiwww.unige.ch/~vision/Viper/`

Texture features Gabors have been applied to texture classification and segmentation, as well as more general vision tasks [9, 21]. We employ a bank of real, circularly symmetric Gabors, defined by

$$f_{mn}(x,y) = \frac{1}{2\pi\sigma_m^2} e^{-\frac{x^2+y^2}{2\sigma_m^2}} \cos(2\pi(u_{0_m} x\cos\theta_n + u_{0_m} y\sin\theta_n)), \qquad (1)$$

where m indexes filter scales, n their orientations, and u_{0_m} gives the centre frequency. The half peak radial bandwidth is chosen to be one octave, which determines σ_m. The highest centre frequency is chosen as $u_{0_1} = 0.5$, and $u_{0_{m+1}} = u_{0_m}/2$. Three scales are used. The four orientations are: $\theta_0 = 0, \theta_{n+1} = \theta_n + \pi/4$. The resultant bank of 12 filters gives good coverage of the frequency domain, and little overlap between filters. The mean energy of each filter is computed for each 16×16 block in the image. This is quantized into 10 bands. A feature is stored for each filter with energy greater than the lowest band. Of the 27648 such possible features for a 256×256 image, an image has at most 3072. Histograms of the mean filter outputs are used to represent global texture characteristics.

3.2 Techniques derived from text retrieval

Inverted files An IF contains an entry for every possible feature consisting of a list of the items which contain that feature. The TR community has developed techniques for building and searching IFs very efficiently [22]. In evaluating a query, only images which contain features present in the query are retrieved. Coupled with appropriate weighting schemes this results in asymmetric similarity measures, in better accord with the psychophysical data (see §2.2).

Feature weighting and relevance feedback As discussed in §2.3, RF can produce a query which better represents a user's information need. We investigate the application of weighting functions used in TR to CBIR. The weighting function can depend upon the *term frequency* tf_j and *collection frequency* cf of the feature, as well as its type (block or histogram). The motivation for using tf and cf is very simple: features with high tf characterize an image well; features with high cf do not distinguish that image well from others [2]. We consider a query q containing N images i with relevances $R_i \in [-1, 1]$. The frequency of feature j in the pseudo-image corresponding to q is[2]

$$tf_{qj} = \frac{1}{N} \sum_{i=1}^{N} tf_{ij} \cdot R_i. \qquad (2)$$

The weighting functions defined in Equations 5 – 9 are derived from typical TR term weighting functions [2]. Some modifications were necessary since the

[2] In this paper, only single-level, positive feedback is used: $R_i = 1$ for all images in q.

image features used can not always be treated in the same way as words in documents. All weighting functions make use of a base weight

$$wf^0_{kqj} = \begin{cases} tf_{qj} & \text{for block features} \\ \operatorname{sgn}(tf_{qj}) \cdot \min\left\{\operatorname{abs}\left(tf_{qj}\right), tf_{kj}\right\} & \text{for histogram features} \end{cases}. \quad (3)$$

(The second case is a generalized histogram intersection.) Two different logarithmic factors are used, which depend upon cf:

$$lcf_{1j} = \begin{cases} \log(\frac{1}{cf_j}) & \text{block} \\ 1 & \text{hist.} \end{cases} \qquad lcf_{2j} = \begin{cases} \log(\frac{1}{cf_j} - 1 + \epsilon) & \text{block} \\ 1 & \text{hist.} \end{cases}. \quad (4)$$

ϵ is added to avoid overflows. The weighting functions investigated are

$$\text{best weighted probabilistic:} \quad wf^1 = wf^0_{kqj} \cdot \left(0.5 + \frac{0.5 tf_{kj}}{\max_j tf_{kj}}\right) \cdot lcf_{2j} \quad (5)$$

$$\text{classical idf:} \quad wf^2 = wf^0_{kqj} \cdot \left(lcf_{1j}\right)^2 \quad (6)$$

$$\text{binary term independence:} \quad wf^3 = wf^0_{kqj} \cdot lcf_{2j} \cdot \quad (7)$$

$$\text{standard tf:} \quad wf^4 = \frac{wf^0_{kqj}}{\sqrt{\sum_m tf^2_{km}}} \cdot \begin{cases} tf_{kj} \cdot tf_{qj} & \text{block} \\ 1 & \text{hist.} \end{cases} \quad (8)$$

$$\text{coordination level:} \quad wf^5 = wf^0_{kqj} \quad (9)$$

For each image k, using weighting method l, a score s^l_{kq} is calculated:

$$s^l_{kq} = \sum_j wf^l_{kqj}. \quad (10)$$

4 Experiments

The performance of *Viper* was evaluated using a set of 500 unconstrained colour images provided by Télévision Suisse Romande. Ten images were selected as queries. Five users then examined all 500 images to determine relevant sets for each query.[3] Neither the number of images to choose nor the similarity criteria were specified. Each query image was presented to *Viper* and the top 20 ranked images were returned. Using a "consistent user" assumption, the relevant set for each user for this query was inspected and the set of relevant images present in the top 20 was then submitted as a second, relevance feedback query. This was done for the five weighting schemes (Equations 5–9), meaning that 300 relevance feedback queries were performed.

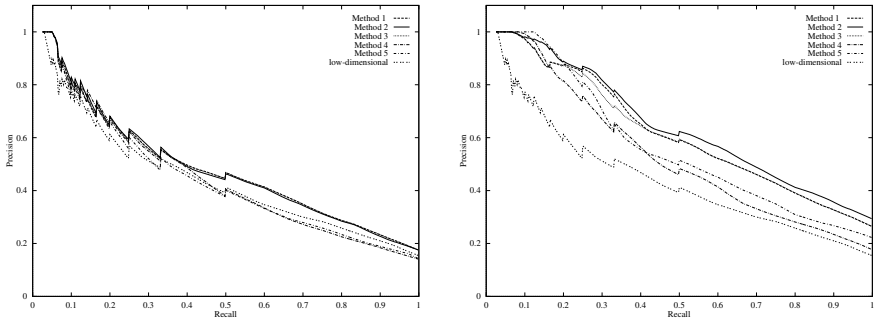
The performance of *Viper* was compared with that of a low-dimensional system of the sort commonly used in image retrieval. The system uses a set of 16 colour, segment, arc and region statistics [23].

[3] All users were computer vision researchers, so some bias can be expected.

System performances are compared using precision P and recall R,

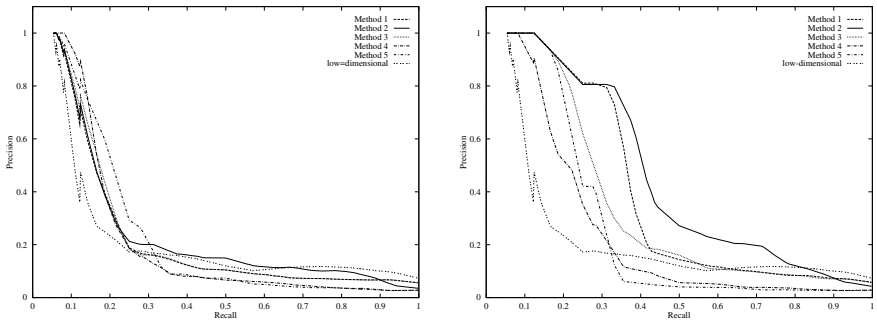
$$P = \frac{r}{N} \qquad\qquad R = \frac{r}{TotRel}, \tag{11}$$

where N is the number of images retrieved, r is the number of relevant images retrieved, and $TotRel$ is the total number of relevant images in the collection. In general, precision decreases as more images are retrieved. An ideal P vs. R graph has $P = 1 \ \forall \ R$.



(a) before feedback (b) after feedback

Fig. 1. Performance of weighting methods averaged over all users and queries.



(a) before feedback (b) after feedback

Fig. 2. Performance of weighting methods on a "hard" query, averaged across users.

Figure 1 shows the performance of the weighting methods averaged over all users and queries. RF improves performance in every case, at all values of recall. This is significant since in general not all relevant images are present in the

top 20 after the initial query: the system is not simply returning those images marked relevant with higher rankings.

System performance varied greatly depending on the nature of the query. Some queries are "easy", in that simple visual features characterize the relevant set. The relevant sets were very similar in these cases, and performance after RF was often perfect ($P = 1 \; \forall \; R$). Figure 2 shows the performance of *Viper* on a "hard" query: an indoor scene labeled by one user as "parliament". The relevant sets for this query varied greatly in size and composition. The effect of RF is even more dramatic in this case. This is to be expected, since no fixed similarity measure can cope with different relevant sets across users.

The best weighting function for this query is method 2 (Equation 6), and this is also the best method averaged over all queries. This is a classic $tf \cdot \log 1/cf$ weight, which has been shown to have information theoretic motivation [2].

5 Conclusion

We have shown how techniques used in TR (inverted files, relevance feedback and term weighting) can be adapted for use in CBIR. IFs permit the use of very large feature spaces, and experiments show that term weighting and RF result in a system which outperforms a low-dimensional vector-space system at every level of recall.

6 Acknowledgments

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

References

1. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. J. of the Am. Soc. for Information Science 41(4) (1990):288–287
2. Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Information Processing and Management 24(5) (1988):513–523
3. Salton, G.: The state of retrieval system evaluation. Information Processing and Management 28(4) (1992):441–450
4. Pun, T., Squire, D. M.: Statistical structuring of pictorial databases for content-based image retrieval systems. Pattern Recognition Letters 17 (1996):1299–1310
5. Niblack, W., Barber, R., Equitz, W., Flickner, M. D., Glasman, E. H., Petkovic, D., Yanker, P., Faloutsos, C., Taubin, G.: QBIC project: querying images by content, using color, texture, and shape. In: Niblack, W., ed., *Storage and Retrieval for Image and Video Databases*, vol. 1908 of *SPIE Proc.* (Apr. 1993), 173–187
6. Jain, A. K., Vailaya, A.: Image retrieval using color and shape. Pattern Recognition 29(8) (Aug. 1996):1233–1244
7. Smith, J. R., Chang, S.-F.: Tools and techniques for color image retrieval. In: Sethi, I. K., Jain, R. C., eds., *Storage & Retrieval for Image and Video Databases IV*, vol. 2670 of *IS&T/SPIE Proceedings*. San Jose, CA, USA (Mar. 1996), 426–437

8. Sclaroff, S., Taycher, L., La Cascia, M.: ImageRover: a content-based browser for the world wide web. In: *IEEE Workshop on Content-Based Access of Image and Video Libraries*. San Juan, Puerto Rico (Jun. 1997), 2–9
9. Ma, W., Manjunath, B.: Texture features and learning similarity. In: CVPR'96 [24], 425–430
10. Pentland, A., Picard, R. W., Sclaroff, S.: Photobook: Tools for content-based manipulation of image databases. Intl. J. of Computer Vision 18(3) (Jun. 1996):233–254
11. Sclaroff, S.: Deformable prototypes for encoding shape categories in image databases. Pattern Recognition 30(4) (Apr. 1997):627–642. (special issue on image databases)
12. Cohen, S. D., Guibas, L. J.: Shape-based image retrieval using geometric hashing. In: *Proc. of the ARPA Image Understanding Workshop* (May 1997), 669–674
13. Ze Wang, J., Wiederhold, G., Firschein, O., Xin Wei, S.: Wavelet-based image indexing techniques with partial sketch retrieval capability. In: *Proc. of the 4th Forum on Research and Technology Advances in Digital Libraries*. Washington D.C. (May 1997), 13–24
14. Carson, C., Belongie, S., Greenspan, H., Malik, J.: Region-based image querying. In: *Proc. of the 1997 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '97)*. San Juan, Puerto Rico (Jun. 1997)
15. Ma, W. Y., Deng, Y., Manjunath, B. S.: Tools for texture- and color-based search of images. In: Rogowitz, B. E., Pappas, T. N., eds., *Human Vision and Electronic Imaging II*, vol. 3016 of *SPIE Proc.*. San Jose, CA (Feb. 1997), 496–507
16. Mokhtarian, F., Abbasi, S., Kittler, J.: Efficient and robust retrieval by shape content through curvature scale space. In: Smeulders and Jain [25], 35–42
17. Tversky, A.: Features of similarity. Psychological Rev. 84(4) (Jul. 1977):327–352
18. Squire, D. M.: Learning a similarity-based distance measure for image database organization from human partitionings of an image set. In: *Proc. of the 4th IEEE Workshop on Applications of Computer Vision (WACV'98)*. Princeton, NJ, USA (Oct. 1998), 88–93
19. Huang, J., Kumar, S. R., Mitra, M.: Combining supervised learning with color correlograms for content-based image retrieval. In: *Proc. of The Fifth ACM Intl. Multimedia Conf. (ACM Multimedia 97)*. Seattle, USA (Nov. 1997), 325–334
20. Rui, Y., Huang, T. S., Ortega, M., , Mehrotra, S.: Relevance feedback: A power tool in interactive content-based image retrieval. IEEE Trans. on Circuits and Systems for Video Technology 8(5) (Sep. 1998):644–655
21. Jain, A., Healey, G.: A multiscale representation including opponent color features for texture recognition. IEEE Trans. on Image Processing 7(1) (Jan. 1998):124–128
22. Witten, I. H., Moffat, A., Bell, T. C.: Managing gigabytes: compressing and indexing documents and images. Van Nostrand Reinhold, 115 Fifth Avenue, New York, NY 10003, USA (1994)
23. Squire, D. M., Pun, T.: A comparison of human and machine assessments of image similarity for the organization of image databases. In: Frydrych *et al.* [26], 51–58
24. Proc. of the 1996 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '96), San Francisco, California (Jun. 1996)
25. Smeulders, A. W. M., Jain, R., eds.: Image Databases and Multi-Media Search, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands (Aug. 1996)
26. Frydrych, M., Parkkinen, J., Visa, A., eds.: The 10th Scandinavian Conf. on Image Analysis (SCIA'97), Lappeenranta, Finland (Jun. 1997)