Content-based query of image databases, inspirations from text retrieval: inverted files, frequency-based weights and relevance feedback

SQUIRE, David, *et al.*

Abstract

In this paper we report the application of techniques inspired by text retrieval research to the content-based query of image databases. In particular, we show how the use of an inverted file data structure permits the use of a feature space of $\mathcal{O}(104)$ dimensions, by restricting search to the subspace spanned by the features present in the query. A suitably sparse set of colour and texture features is proposed. A scheme based on the frequency of occurrence of features in both individual images and in the whole collection provides a means of weighting possibly incommensurate features in a compatible manner, and naturally extends to incorporate relevance feedback queries. The use of relevance feedback is shown consistently to improve system performance, as measured by precision and recall.

Reference

SQUIRE, David, *et al.* 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*. 1999. p. 143-149

Content-based query of image databases, inspirations from text retrieval: inverted files, frequency-based weights and relevance feedback

David McG. Squire, Wolfgang Müller, Henning Müller, Jilali Raki
Vision Group, Computer Science Department,
University of Geneva, 24 rue Général Dufour,
1211 Genève 4, Switzerland

Abstract

In this paper we report the application of techniques inspired by text retrieval research to the content-based query of image databases. In particular, we show how the use of an inverted file data structure permits the use of a feature space of $\mathcal{O}(10^4)$ dimensions, by restricting search to the subspace spanned by the features present in the query. A suitably sparse set of colour and texture features is proposed. A scheme based on the frequency of occurrence of features in both individual images and in the whole collection provides a means of weighting possibly incommensurate features in a compatible manner, and naturally extends to incorporate relevance feedback queries. The use of relevance feedback is shown consistently to improve system performance, as measured by precision and recall.

1 Introduction

In recent years, the use of digital image collections has become common, both on the world wide web and in the preparation of both electronic and paper publications. The need for tools to manage this rapidly-increasing quantity of visual data is greater than ever. Specifically, there is great interest in systems which allow users to query image databases. The attachment of text labels to images is inadequate, since identical images can be described in different ways, and controlled vocabulary indexing is now considered insufficient even in text retrieval systems. Consequently, there is great interest in Content-Based Image Retrieval Systems (CBIRSs).

The content-based retrieval of text documents has been studied for more than forty years. Many of the insights and techniques of text retrieval have been ignored by image retrieval researchers, or reinvented without the benefit of knowledge of the prior work. *Relevance Feedback* was introduced for improving text retrieval performance more than thirty years ago, and its utility is long-established [27]. Similarly, a great variety of term-weighting approaches have been investigated, both empirically and theoretically [9, 26]. Means of system evaluation have also been thoroughly studied [25], yet *Precision* and *Recall* [25, 35], the usual performance measures, are ignored by many researchers.

Text retrieval systems usually treat each possible term (*i.e.* word) as a dimension of the search space. Consequently, spaces with $\mathcal{O}(10^4)$ dimensions are typical. It has thus been necessary to develop techniques for efficient search in such spaces, which is clearly beyond the scope of standard spatial indexing techniques. 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. This is not the same as having a numerical value of zero for many attributes: a value of zero for numerical colour feature contains information about image appearance and must be evaluated in the processing of a query; the *absence* of a feature does not. The search is thus restricted to the subspace spanned by the terms of the query. The data structure which makes such a subspace search efficient is the *Inverted File*.

Conversely, considerable effort has been devoted by image retrieval researchers to the search for compact image representations (choosing the "right" features), and to the use of techniques such as factor analysis [24] or self-organizing maps [11] to reduce the feature space dimensionality, so that search can be performed using techniques such as *k*-*d* trees and R-trees [40].

In this paper we present an image retrieval system which uses an inverted file, with more than 80000 possible features per image. A typical image contains $\mathcal{O}(10^3)$ features. A feature weighting scheme based on the frequencies of features both in the query image and also in the entire collection, commonly used in text retrieval, is employed. A relevance feedback scheme is also used. Evaluation using precision and recall demonstrates a clear improvement over a previously-reported system using a smaller feature set and nearest-neighbour search.

2 Related Work

The aim of a CBIRS is to retrieve images from a database based on their *similarity* to a query image or sketch [10, 44]. It is acknowledged that the general computer vision problem remains unsolved: semantic retrieval is still impossible. Some attempt to avoid the problem by restricting attention to images from restricted domains, such as industrial trademarks [13] or marine animals [19]. Others tackle the general problem: low-level features are extracted from the images, and an attempt is made to capture image similarity using some function of these features. Object recognition is not attempted.

2.1 Features

2.1.1 Colour

By far the most commonly used feature is colour (for example, [13, 20, 32]). Often only the *global* colour properties of images are used. These are usually computed in a colour space thought to correspond to human perception of colour differences (*e.g.* HSV [32, 37] or CIE [30]). The most commonly used representation of these properties is the colour histogram [10, 37]. *Histogram intersection*, which defines the similarity between two histograms as the (normalized) number of pixels in common for each pair of bins, is the usual measure of the distance between colour representations. A disadvantage of this measure is that it takes no account of the perceptual similarity between bins [37]. Measures exist which use a matrix of bin similarity coefficients [20], but the choice of coefficients is not obvious, and the cost is quadratic.

2.1.2 Texture

Many systems use texture to improve image characterization (for example [17, 18, 21]). There is a great variety of texture features available to system designers: hierarchies of Gabor filters [16]; the Wold features [15] used in MIT's Photobook system [22]; the coarseness, contrast, and directionality features used in IBM's QBIC [20]; wavelet-based decompositions [43]; and many more. When such features are global, they share the drawbacks of the global colour features discussed above.

2.1.3 Shape

The third class of features that appear frequently in current CBIRSs is based on shape. These features are again generally global: each image is assumed to contain a single shape. This restriction mean that shape features are most easily applied to images drawn from restricted domains. A good example is modal matching, a deformation-based method, which has been applied to isolated fish, rabbits and machine tools [29]. Other shape-based approaches include multi-scale representation of curves [1]; histograms of edge directions, which have been applied to trademarks [13]; and matching to templates of shape components such as corners, line segments or circular arcs [4].

2.1.4 Local features

Numerous researchers have recognized that global features are inadequate for many image retrieval tasks. Users are often interested in the spatial layout of the colours, textures and shapes in an image, and may be interested only in certain objects. Some have addressed this problem by seeking features which retain spatial information as well as other image properties, such as wavelet decompositions [38, 44]. Wavelet decompositions are usually extremely vulnerable to small shifts of objects in the image. An alternative approach to extracting local information is to segment the image into regions, and then extract features of those regions [2, 7, 17, 32]. Features such as color and texture are extracted for each region, as well as spatial properties such as size, location and relationships to other regions. This approach turns the image retrieval problem into the non-trivial problem of labeled graph matching.

2.2 Similarity

CBIRSs aim to return images which are similar to an example image, sketch or collection of regions. Remarkably, the meaning of "similarity" in this context is rarely addressed. Those who do address it discover its difficulty, *e.g.* "the results of the subjective test indicated that human judgments of shape similarity noticeably differ" [19, p. 38]. Similarity between images is typically defined using the distance between image points in a multidimensional feature space, as given by some metric: images close to the query are "similar" to the query. The aim of such systems, however, is to return images that are similar to the query according to the user's perception. The fact that these two notions of similarity may be very different is rarely discussed. It is often implied that if one chooses the "right" features (an appropriate colour space [30, 37], texture features "corresponding to human perception" [15]), then proximity in feature space *must* correspond to perceptual similarity.

There are several reasons to doubt this, the most fundamental being the *metric assumption*. There is psychophysical evidence that human similarity judgments do not obey the requirements of a metric. Specifically: "[Self-identity] is somewhat problematic, symmetry is apparently false, and the triangle inequality is hardly compelling" [36, p. 329]. For image retrieval, the lack of symmetry is the most important issue. In essence, the features which are significant in computing similarity depend on which of a pair of items is the query: the variant is more similar to the prototype than vice versa. There have been some preliminary attempts to apply Tversky's set theoretic similarity functions to CBIRSs [28], but the psychophysical literature on similarity seems to have been largely ignored by CBIRS researchers.

Some authors have addressed the fact that feature space distance is not necessarily equivalent to perceptual similarity. Self-organizing maps have been used to cluster texture features according to class labels provided by human users [16]. Distance Learning Networks have been used to attempt to learn a mapping from feature space to "perceptual similarity space" using human similarity judgment data [33].

3 *Viper* System Overview

Viper, inspired by text retrieval systems, uses a very large number of simple features.[1] The present version employs both local and global image colour and spatial frequency features, extracted at several scales, and their frequency statistics in both images 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.

The fundamental difference between traditional computer vision and image database applications is that there is a human "in the loop". The system is provided with low-level features, and interaction with the user via relevance judgments allows a combination of these features to be discovered which corresponds to the user's desires. In effect, a simple image classifier is learnt "on the fly", according to the user's feedback.

More than 80000 features are available to the system, such as the mode colour of various regions, or the quantized average energies of the outputs of Gabor filters at a various orientations and scales. Each image has $\mathcal{O}(10^3)$ such features, the mapping from features to images being stored in an inverted file.

3.1 Inverted Files

Inverted files are the most common data structure used in text retrieval. An inverted file contains an entry for every possible feature (term) which consists of a list of the images (documents) which contain that feature, the frequency of occurrence of that feature in the collection, and possibly the frequency of that feature in each image. The text retrieval community has developed techniques for building and searching inverted files very efficiently [41].

Restricting the search to the subspace of the query, coupled with an appropriate weighting scheme, results in asymmetric similarity measures, in accordance with the psychophysical data discussed in §2.2.

3.2 Colour Features

It is desirable that the colour space used in an image retrieval system should be "perceptually" uniform, meaning that small changes in the colour coordinates should correspond to small perceptual differences. The *RGB* space does not have this property. The *HSV* colour space offers improved perceptual uniformity, and is easier to compute and invert than systems such as *CIE-LUV* or *CIE-LAB* [32].

Viper uses a palette of 166 colours, derived by uniformly quantizing the cylindrical *HSV* colour space into 18 hues, 3 saturations, and 3 values. These are augmented by 4 grey levels. This choice of quantization means that more tolerance is given to changes in saturation and value, which is desirable since these channels can be effected by lighting conditions and viewpoint.

Two sets of features are then extracted from the quantized HSV image. The first is equivalent to a conventional colour histogram, with the variation that bins containing zero pixels are discarded. There are thus 166 possible colour histogram features, of which most images contain only about 40.

The second class of features represent local colour properties of the image. The image is divided into square blocks at four scales, ranging from 16×16 through to 128×128. The mode colour is calculated for each block. The occurrence of of a given color in a particular block 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.

3.3 Texture Features

Two dimensional Gabor filters have frequently been proposed as a framework for describing and understanding the orientation- and frequency-selective properties of neurons in the visual cortex [6], and banks of Gabor filters have often been applied to texture classification and segmentation [14, 39], as well as more general vision tasks [3, 12, 16]. We employ a bank of real, circularly symmetric Gabor filters, defined in the spatial domain by

$$\begin{aligned} f_{mn}(x,y) &= \tfrac{1}{2\pi\sigma_m^2} e^{-\frac{x^2+y^2}{2\sigma_m^2}} \cos(2\pi(u_{0_m} x \cos\theta_n \\ &\quad + u_{0_m} y \sin\theta_n)), \end{aligned} \tag{1}$$

where m indexes the scales of the filters, and n their orientations. The centre frequency of the filter is specified by u_{0_m}. The half peak radial bandwidth is given by

$$B_r = \log_2 \left(\frac{2\pi\sigma_m u_{0_m} + (2\ln 2)^{1/2}}{2\pi\sigma_m u_{0_m} - (2\ln 2)^{1/2}} \right) \tag{2}$$

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

(after [12]). B_r is chosen to be 1 (*i.e.* a bandwidth of one octave), which then allows us to compute σ_m:

$$\sigma_m = \frac{3(2\ln 2)^{1/2}}{2\pi u_{0_m}}. \tag{3}$$

The highest centre frequency is chosen as $u_{0_1} = \frac{0.5}{1+\tan(1/3)} \approx 0.5$ so that it is within the discrete frequency domain. The centre frequency is halved at each change of scale, which implies that σ is doubled (Equation 3). The orientation of the filters varies in steps of $\pi/4$, and three scales are used. These choices result in a bank of 12 filters which gives good coverage of the frequency domain, and little overlap between filters [12]. For practical implementation, filters are truncated at 3σ, giving kernels of sizes 9×9, 17×17 and 35×35.

The use of circularly symmetric filters means that Equation 2 is separable. The 2-D convolution can thus be computed using four 1-D convolutions, which reduces the number of computations required for an $N \times N$ kernel by a factor of order N [12].

These filters are applied to the image, and the mean energy of each filter is computed for each 16×16 block in the image. The energy is then quantized into 10 bands, which were chosen by examining histograms of the filter energy at each pixel for 500 images. A feature is stored for each filter which has an energy in a band greater than the $[0, 2)$ band. This means that there are 27648 possible such features for a 256×256 image, of which a given image may have at most 3072 (in practice this does not arise). Histograms of the mean filter outputs are also stored, giving a measure of the global texture characteristics of the image.

3.4 Similarity Computation and Relevance Feedback

Relevance feedback has been shown to be extremely useful in text retrieval applications [27], and it has been applied in some CBIRSs [42]. In an image database application, it offers two advantages. First, by augmenting the query with features from relevant retrieved images, one can produce a query which better represents the user's information need. The second advantage is unique to the image retrieval problem. In text retrieval, feature extraction is free: the documents' component words are themselves the features. This is not the case in image retrieval. We envisage a system in which expensive features are extracted from images when the database is built, even though such features may be too numerous and too expensive to evaluate for a new query image. Nevertheless, once some images are retrieved using a subset of cheap features, complex features can be added to the query via relevance feedback. The use of inverted file data structure means that the addition of "rare" features adds to the cost of query evaluation only when they are likely to be relevant.

The similarity between images in the database and a query is always computed as if it were a relevance feedback query – the first query simply consists of a single relevant image. Features are combined according to by summing their frequencies of occurrence *df* (block features have a frequency of 1). For a query q containing N images i with relevance levels $R_i \in [-1, +1]$ and features j with frequencies df_{ij}, we have

$$df_{qj} = \frac{1}{N} \sum_{i=1}^{N} df_{ij} \cdot R_i \tag{4}$$

Features can be evaluated in order according to df_{qj}, allowing the search to be pruned.

For each feature j of the M features in q, the list of images containing that feature is retrieved and added to the pool of candidate images. For non-histogram features, the score s_k of each image k is updated according to

$$s_{k_{new}} = s_{k_{old}} + df_{qj} df_{kj} \log cf_j^{-1}, \tag{5}$$

where cf_j is the frequency of the feature j in the entire database. This equation originates from a text retrieval scheme. Its motivation is very simple: features which are common in an image characterize that image well; features which are common in the collection do not distinguish that image well from others [26]. It is worth noting that dimensionality reduction schemes such as Principal Components Analysis [23, 24] can have the effect of eliminating these "rare" dimensions which can in fact be very useful for creating a specific query. For histogram features, the score is updated according to

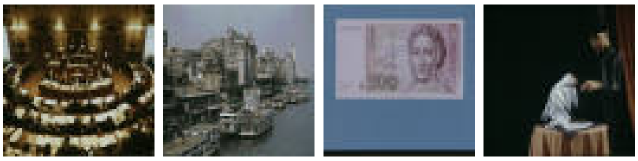
$$s_{k_{new}} = s_{k_{old}} + \mathrm{sign}(df_{qj}) \min(|df_{qj}|, df_{kj}) \log cf_j^{-1}, \tag{6}$$

which is a weighted variant of standard histogram intersection.

4 Experiments

An evaluation of *Viper* performance was carried out using a set of 500 unconstrained colour images provided by Télévision Suisse Romande. These images contained some "runs" of images from the same footage, so some highly similar images are present. Sample images are shown in Figure 1.

Ten images were selected as queries by a human user, who examined the whole set of 500 images to determine the set of relevant images for each query. These relevant sets varied greatly in size, the smallest containing three images and the largest nineteen (which partially accounts for the differing appearances of the graphs in



"Parliament" "Cityscape" "German Banknote" "Magic Show"

Figure 1: Sample images from the *Viper* test database.

Figures 2 and 3). The degree of visual similarity also differed greatly, ranging from 7 very similar standard images of German banknotes (the collection contains 37 such images of various countries' banknotes) to set a of quite different photographs taken in various libraries. It is usual to get a group of users to perform this task and then to pool the results, though this makes it more difficult to evaluate relevance feedback experiments, since different users may have differing notions of what is relevant.

Each image was presented to *Viper* as an initial query, and 20 images were returned to the user. The user could then mark as many as desired of these images as relevant. Negative feedback was not employed in this experiment. This set of relevant images was then submitted as a second query.

A comparison was made between the performance of *Viper* and that of a vector space system of the sort commonly used in image retrieval, reported in earlier work [34]. The system uses a set of 16 colour, segment, arc and region statistics. The Euclidean distance between each image in the database and each query image is computed exhaustively.

System performances are compared using precision and recall, which are defined as

$$\begin{aligned} \mathit{Precision} &= \frac{r}{N} \qquad (7) \\ \mathit{Recall} &= \frac{r}{R}, \qquad (8) \end{aligned}$$

where N is the total number of images (documents) retrieved, r is the number of relevant images retrieved, and R is the total number of relevant images in the collection.

Precision and recall data are often presented in the form of a *Precision* vs. *Recall* graph, which shows, in general, how precision decreases as increasingly large fractions of the collection are retrieved. An ideal *Precision* vs. *Recall* graph has *Precision* = 1 for all values of *Recall*: all the relevant images are retrieved before any irrelevant ones. The closer *Precision* stays to 1, the better.

Figures 2 and 3 show the performances of the two systems on six of the ten queries. Two plots are shown

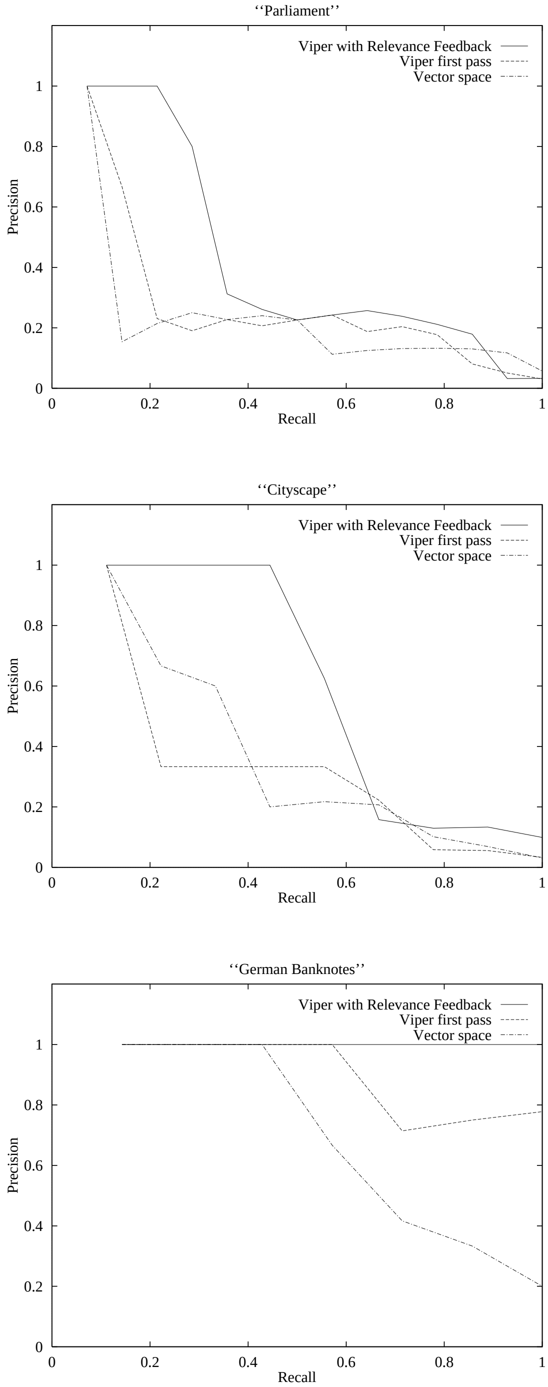


Figure 2: Precision vs. Recall graphs for 3 queries, comparing the vector space system with the Viper system, both with and without one phase of relevance feedback.

for the *Viper* system, indicating performance before and after the relevance feedback step. It should be remembered that the user was only shown the top twenty ranked images after the first pass: it was thus not in general possible to include all relevant images in the

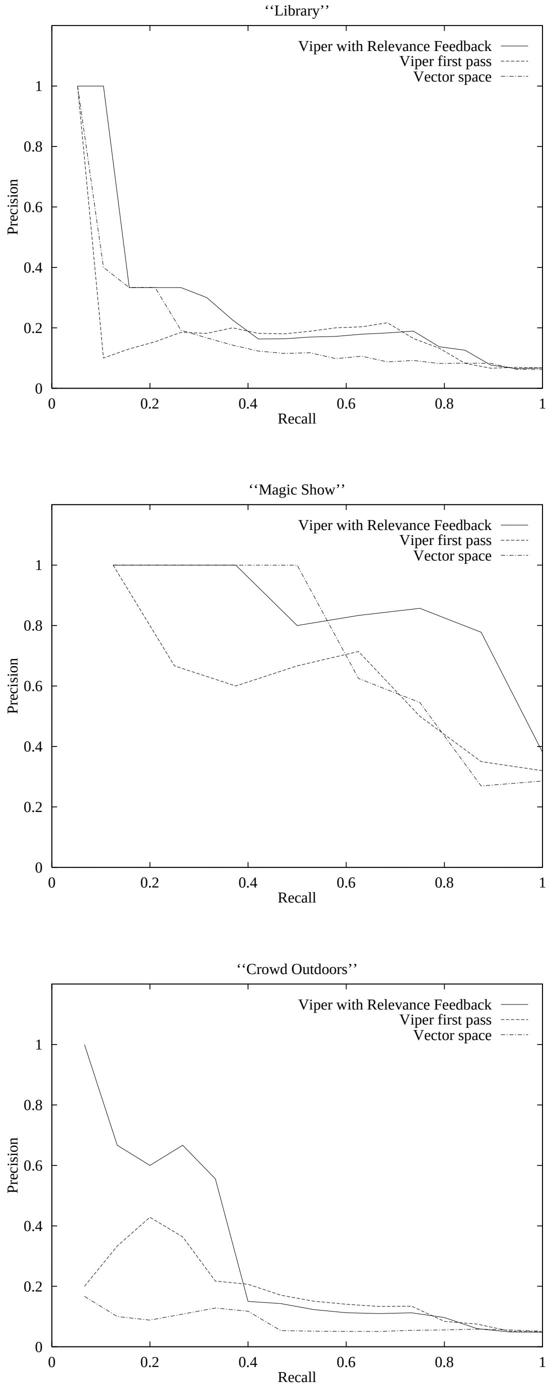


Figure 3: Precision vs. Recall graphs for 3 queries, comparing the vector space system with the Viper system, both with and without one phase of relevance feedback.

feedback query.

The plots clearly indicate the value of relevance feedback. In all cases except the "Crowd Outdoors" query the use of relevance feedback resulted in an improvement in *Precision* to a value of 1 over a large part of the *Recall* domain. The *Precision* of the relevance feedback queries remains higher than that of either the first pass of *Viper* or that of the vector space system at almost all *Recall* values, often dramatically so.

The performance of the very large feature space *Viper* first pass is also better than that of the vector space system for most queries, the exceptions being the broad categories of "Cityscape" and "Magic Show". In both cases the relevance feedback phase reversed this situation.

5 Conclusion

In this paper we have indicated how techniques inspired by text retrieval can be applied to the content-based query of image databases. We believe that there is much to be learnt from the decades of research in text retrieval, despite the fact that the terms of text queries (words) are much closer to the semantic level than the simple features usually used for image retrieval.

The use of inverted files, coupled with an appropriate choice of discrete features, allows feature spaces of extremely high dimensionalities to be searched efficiently. We have demonstrated the application of this technique to an image retrieval system with more than 80000 possible features.

The use of *Precision* and *Recall* graphs provides a standard means of comparing system performances. Experiments using 10 queries on a test database of 500 images demonstrated that the *Viper* system, using frequency-based weights, performed better than a vector space system even without relevance feedback. One iteration of relevance feedback always improved performance, often dramatically.

Acknowledgments

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

References

[1] A. D. Bimbo and P. Pala. Shape indexing by multiscale representation. In Smeulders and Jain [31], pages 43–50.

[2] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Region-based image querying. In *Proceedings of the 1997 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '97)*, San Juan, Puerto Rico, June 1997. IEEE Computer Society.

[3] C. Chang and S. Chatterjee. Ranging through Gabor logons, a consistent, hierarchical approach.

IEEE Transactions on Neural Networks, 4(5):827–843, September 1993.

[4] S. D. Cohen and L. J. Guibas. Shape-based image retrieval using geometric hashing. In *Proceedings of the ARPA Image Understanding Workshop*, pages 669–674, May 1997.

[5] *Proceedings of the 1996 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '96)*, San Francisco, California, June 1996.

[6] J. G. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1148–1161, 1993.

[7] A. Dimai. Spatial encoding using differences of global features. In I. K. Sethi and R. C. Jain, editors, *Storage and Retrieval for Image and Video Databases V*, volume 3022 of *SPIE Proceedings*, pages 352–360, February 1997.

[8] M. Frydrych, J. Parkkinen, and A. Visa, editors. *The 10th Scandinavian Conference on Image Analysis (SCIA'97)*, Lappeenranta, Finland, June 1997. Pattern Recognition Society of Finland.

[9] W. R. Greiff. A theory of term weighting based on exploratory data analysis. In W. B. Croft, A. Moffat, C. J. van Rijsbergen, R. Wilkinson, and J. Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 11–19, Melbourne, Australia, August 1998. ACM Press, New York.

[10] A. Gupta and R. Jain. Visual information retrieval. *Communications of the ACM*, 40(5):70–79, May 1997.

[11] K. Han and S.-H. Myaeng. Image organization and retrieval with automatically constructed feature vectors. In H.-P. Frei, D. Harman, P. Schäuble, and R. Wilkinson, editors, *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96)*, pages 157–165, Zürich, Switzerland, August 1996.

[12] A. Jain and G. Healey. A multiscale representation including opponent color features for texture recognition. *IEEE Transactions on Image Processing*, 7(1):124–128, January 1998.

[13] A. K. Jain and A. Vailaya. Image retrieval using color and shape. *Pattern Recognition*, 29(8):1233–1244, August 1996.

[14] T. Kuyel and J. Ghosh. A fast space localized computation of the outputs of a Gabor filter bank. In *Proceedings of the IASTED Conference on Signal and Image Processing (SIP'95)*, pages 511–514, Las Vegas, USA, November 1995.

[15] F. Liu and R. Picard. Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):722–733, July 1996.

[16] W. Ma and B. Manjunath. Texture features and learning similarity. In CVPR'96 [5], pages 425–430.

[17] W. Y. Ma, Y. Deng, and B. S. Manjunath. Tools for texture- and color-based search of images. In B. E. Rogowitz and T. N. Pappas, editors, *Human Vision and Electronic Imaging II*, volume 3016 of *SPIE Proceedings*, pages 496–507, San Jose, CA, February 1997.

[18] T. P. Minka and R. W. Picard. Interactive learning using a "society of models". In CVPR'96 [5], pages 447–452.

[19] F. Mokhtarian, S. Abbasi, and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. In Smeulders and Jain [31], pages 35–42.

[20] W. Niblack, R. Barber, W. Equitz, M. D. Flickner, E. H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. QBIC project: querying images by content, using color, texture, and shape. In W. Niblack, editor, *Storage and Retrieval for Image and Video Databases*, volume 1908 of *SPIE Proceedings*, pages 173–187, April 1993.

[21] S. C. Orphanoudakis, C. E. Chronaki, and D. Vamvaka. I^2net: Content-based similarity search in geographically distributed repositories of medical images. *Computerized Medical Imaging and Graphics*, 20(4):193–207, 1996.

[22] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook: Tools for content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254, June 1996.

[23] Z. Pečenović. Image retrieval using latent semantic indexing. Final year graduate thesis, Audio-Visual Communications Lab, Ecole Polytechnique Fédérale de Lausanne, Switzerland, June 1997.

[24] T. Pun and D. M. Squire. Statistical structuring of pictorial databases for content-based image retrieval systems. *Pattern Recognition Letters*, 17:1299–1310, 1996.

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

[26] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

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

[28] S. Santini and R. Jain. Similarity queries in image databases. In CVPR'96 [5], pages 646–651.

[29] S. Sclaroff. Deformable prototypes for encoding shape categories in image databases. *Pattern Recognition*, 30(4):627–642, April 1997. (special issue on image databases).

[30] S. Sclaroff, L. Taycher, and M. La Cascia. ImageRover: a content-based browser for the world wide web. In *IEEE Workshop on Content-Based Access of Image and Video Libraries*, pages 2–9, San Juan, Puerto Rico, June 1997.

[31] A. W. M. Smeulders and R. Jain, editors. *Image Databases and Multi-Media Search*, Kruislaan 403, 1098 SJ Amsterdam, The Netherlands, August 1996. Intelligent Sensory Information Systems, Faculty of Mathematics, Computer Science, Physics and Astronomy, Amsterdam University Press.

[32] J. R. Smith and S.-F. Chang. Tools and techniques for color image retrieval. In I. K. Sethi and R. C. Jain, editors, *Storage & Retrieval for Image and Video Databases IV*, volume 2670 of *IS&T/SPIE Proceedings*, pages 426–437, San Jose, CA, USA, March 1996.

[33] D. M. Squire. Learning a similarity-based distance measure for image database organization from human partitionings of an image set. In *Proceedings of the Fourth IEEE Workshop on Applications of Computer Vision (WACV'98)*, pages 88–93, Princeton, NJ, USA, October 1998.

[34] D. M. Squire and T. Pun. A comparison of human and machine assessments of image similarity for the organization of image databases. In Frydrych et al. [8], pages 51–58.

[35] 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.

[36] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, July 1977.

[37] A. Vellaikal and C.-C. J. Kuo. Content-based image retrieval using multiresolution histogram representation. In C.-C. J. Kuo, editor, *Digital Image Storage and Archiving Systems*, volume 2606 of *SPIE Proceedings*, pages 312–323, Philadelphia, PA, USA, October 1995.

[38] M. Vetterli and J. Kovačević. *Wavelets and subband coding.* Prentice-Hall, Englewood Cliffs, NJ, 1995.

[39] T. P. Weldon and W. E. Higgins. Integrated approach to texture segmentation using multiple gabor filters. In P. Delogne, editor, *IEEE International Conference on Image Processing (ICIP'96)*, pages 955–958, Lausanne, Switzerland, September 1996.

[40] D. A. White and R. Jain. Similarity indexing: algorithms and performance. In I. K. Sethi and R. C. Jain, editors, *Storage and Retrieval for Still Image and Video Databases IV*, volume 2670 of *SPIE Proceedings*, pages 62–73, March 1996.

[41] I. H. Witten, A. Moffat, and T. C. Bell. *Managing gigabytes: compressing and indexing documents and images.* Van Nostrand Reinhold, 115 Fifth Avenue, New York, NY 10003, USA, 1994.

[42] M. E. Wood, N. W. Campbell, and B. T. Thomas. Iterative refinement by relevance feedback in content-based digital image retrieval. In *Proceedings of The Fifth ACM International Multimedia Conference (ACM Multimedia 98)*, pages 13–20, Bristol, UK, September 1998.

[43] R. Zarita and S. Lelandais. Wavelets and high order statistics for texture classification. In Frydrych et al. [8], pages 95–102.

[44] J. Ze Wang, G. Wiederhold, O. Firschein, and S. Xin Wei. Wavelet-based image indexing techniques with partial sketch retrieval capability. In *Proceedings of the Fourth Forum on Research and Technology Advances in Digital Libraries*, pages 13–24, Washington D.C., May 1997.