Helsinki University of Technology
Dissertations in Computer and Information Science
Espoo 2003 Report D1

INTERACTIVE IMAGE RETRIEVAL USING SELF-ORGANIZING MAPS

Markus Koskela

Dissertation for the degree of Doctor of Science in Technology to be presented with due permission of the Department of Computer Science and Engineering for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 14th of November, 2003, at 12 o'clock noon.

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
FINLAND

Distribution:
Helsinki University of Technology
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
FINLAND
Tel. +358-9-451 3272
Fax. +358-9-451 3277
http://www.cis.hut.fi

Available in pdf format at http://lib.hut.fi/Diss/2003/isbn9512267659/

ISBN 951-22-6764-0 (printed version)
ISBN 951-22-6765-9 (electronic version)
ISSN 1459-7020

Otamedia Oy
Espoo 2003

Koskela, M. (2003): **Interactive image retrieval using Self-Organizing Maps**. Doctoral thesis, Helsinki University of Technology, Dissertations in Computer and Information Science, Report D1, Espoo, Finland.

Keywords: Content-based image retrieval (CBIR), Self-Organizing Map (SOM), relevance feedback, MPEG-7.

ABSTRACT

Digital image libraries are becoming more common and widely used as visual information is produced at a rapidly growing rate. Creating and storing digital images is nowadays easy and getting more affordable all the time as the needed technologies are maturing and becoming eligible for general use. As a result, the amount of data in visual form is increasing and there is a strong need for effective ways to manage and process it. In many settings, the existing and widely adopted methods for text-based indexing and information retrieval are inadequate for these new purposes.

Content-based image retrieval addresses the problem of finding images relevant to the users' information needs from image databases, based principally on low-level visual features for which automatic extraction methods are available. Due to the inherently weak connection between the high-level semantic concepts that humans naturally associate with images and the low-level visual features that the computer is relying upon, the task of developing this kind of systems is very challenging. A popular method to improve retrieval performance is to shift from single-round queries to navigational queries where a single retrieval instance consists of multiple rounds of user–system interaction and query reformulation. This kind of operation is commonly referred to as relevance feedback and can be considered as supervised learning to adjust the subsequent retrieval process by using information gathered from the user's feedback.

In this thesis, an image retrieval system named PicSOM is presented, including detailed descriptions of using multiple parallel Self-Organizing Maps (SOMs) for image indexing and a novel relevance feedback technique. The proposed relevance feedback technique is based on spreading the user responses to local SOM neighborhoods by a convolution with a kernel function. A broad set of evaluations with different image features, retrieval tasks, and parameter settings demonstrating the validity of the retrieval method is described. In particular, the results establish that relevance feedback with the proposed method is able to adapt to different retrieval tasks and scenarios.

Furthermore, a method for using the relevance assessments of previous retrieval sessions or potentially available keyword annotations as sources of semantic information is presented. With performed experiments, it is confirmed that the efficiency of semantic image retrieval can be substantially increased by using these features in parallel with the standard low-level visual features.

PREFACE

The research presented in this thesis has been carried out at the Laboratory of Computer and Information Science (CIS) at Helsinki University of Technology during the years 1998-2003. In 1998, I first became in contact with scientific research as a summer student trainee hired to the newly started PicSOM project and my stay at the laboratory has extended ever since. This research has been funded by the Academy of Finland, within the Finnish Centre of Excellence Programme (2000-2005) project *New Information Processing Principles* and by TEKES (The National Technology Agency) with related funding. In addition, the Academy of Finland has funded this work in the project *Neural methods in information retrieval based on automatic content analysis and relevance feedback.* Personal grants from the Finnish Foundation for Promotion of Technology and Emil Aaltonen Foundation are hereby gratefully acknowledged.

First of all, I would like to express my gratitude to my advisor and the manager of our research project, Dr. Jorma Laaksonen, for exquisite daily guidance, teaching me a lot about scientific practice, and bearing the main responsibility for the programming work. I am obliged to my supervisor, Academy Professor Erkki Oja, for lasting encouragement and support throughout my work, and also for expert comments on the contents of this thesis. The manuscript was reviewed by Prof. Irwin King and Dr. Timo Ojala, whose comments and constructive suggestions led to considerable improvements in the thesis.

I am also grateful to all (present and former) members of our research group, especially Dr. Sami Brandt, Mr. Sami Laakso, and Mr. Mika Rummukainen. For an enjoyable working environment, I wish to thank the staff and co-workers at the CIS laboratory, Mr. Ville Könönen and Mr. Mika Inki in particular. For a delightful contrast to work and for providing a wider perspective on life, I wish to thank all my friends.

Finally, I wish to thank my family for all their support and encouragement during my studies.

Otaniemi, October 2003

Markus Koskela

CONTENTS

LIST OF ABBREVIATIONS

BMU	Best matching unit
CBIR	Content-based image retrieval
COIL	Columbia Object Image Library
EM	Expectation-maximization
EMD	Earth mover's distance
GE	Generalized Euclidean distance
GoF, GoP	Group of frames, Group of pictures
GMM	Gaussian mixture model
HI	Histogram intersection
HSV	Hue, saturation, value
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICA	Independent component analysis
ISO	International Organization for Standardization
IR	Information retrieval
JD	Jeffrey divergence
KL	Kullback-Leibler
kNN	k-nearest-neighbor
LBG	Linde-Buzo-Gray
LSI	Latent semantic indexing
MA	Mahalanobis distance
MDS	Multidimensional scaling
MLP	Multilayer perceptron
MPEG	Moving Picture Experts Group
MRML	Multimedia Retrieval Markup Language
PCA	Principal component analysis
QBE, QBPE	Query by (pictorial) example
RBF	Radial basis function
RDF	Resource Description Framework
RGB	Red, green, blue
SAR, MRSAR	(Multi-resolution) simultaneous auto-regressive model
SGI	Silicon Graphics, Inc.
SOM	Self-Organizing Map
SQ	Scalar quantization
SQL	Structured query language
SVD	Singular value decomposition

SVM	Support vector machine
TREC	Text REtrieval Conference
TS-SOM	Tree Structured Self-Organizing Map
VQ	Vector quantization
VSM	Vector space model
W3C	WWW Consortium
WWW	World Wide Web
XM	eXperimentation Model

LIST OF SYMBOLS

$\mathcal{D}$	image database
$\mathcal{D}^{\oplus}, \mathcal{D}^{\ominus}$	relevant and non-relevant images
$\mathcal{D}_n$	images retrieved on nth query round
$\mathcal{D}(n)$	cumulative set of seen images in a query on nth query round
$\mathcal{D}^+_n, \mathcal{D}^-_n$	relevant and non-relevant images seen on nth query round
$\mathcal{D}^+(n), \mathcal{D}^-(n)$	cumulative sets of relevant and non-relevant seen images
$\mathcal{D}'(n)$	yet unseen images in a query on nth query round
$\mathcal{D}^{\alpha}, \mathcal{D}^{\beta}$	intermediate image sets during query processing
N	size of an image database
N^a_b	cardinality of an image set $\mathcal{D}^a_b$, $N^a_b = \#\{\mathcal{D}^a_b\}$
$\mathcal{I}_i$	ith image in a database
$\mathcal{I}^{\oplus}$	target image in target search
$\mathbf{f}^m_i$	mth feature vector of image $\mathcal{I}_i$
$f^m_i(k)$	kth component of $\mathbf{f}^m_i$
K, K_m	dimensionality of (mth) feature space
M	number of parallel features
$D(\mathcal{I}_i, \mathcal{I}_j)$	distance between images $\mathcal{I}_i$ and $\mathcal{I}_j$
$d(\mathbf{f}_i, \mathbf{f}_j)$	distance between images $\mathcal{I}_i$ and $\mathcal{I}_j$ according to feature $\mathbf{f}$
W_m	weight parameter of mth feature space
w_{mk}	weight of kth component of mth feature
$\mathbf{A}$	quadratic-form similarity weight matrix
L	length of the convolution window
l	convolution window parameter
$\mathbf{q}$	query point in a feature space
μ	mean
σ	standard deviation
$\mathbf{\Sigma}$	covariance matrix
$\mathbf{x}$	input sample in SOM training
t	step index in SOM training
$c(\mathbf{x})$	BMU of input sample $\mathbf{x}$
$\mathbf{m}_i$	ith SOM model vector
$h(t; c(\mathbf{x}), i)$	neighborhood function in SOM training
$\mathcal{H}_n$	query history on nth query round
$\mathcal{A}_n$	user action on nth query round
$x_+(n), x_-(n)$	values of positive and negative scores on nth query round
d_{ij}	normalized distance of map units i and j
$\mathbf{X}$	term-by-document matrix
$\mathcal{P}$	retrieval precision
$\mathcal{R}$	retrieval recall
ρ	*a priori* probability of a ground-truth class

1 INTRODUCTION

Producing visual content in digital form is becoming more and more common and affordable. Digital cameras, scanners, multimedia portable phones, and powerful personal computers are already available at reasonable prices. Data storage units have evolved as well as other computer hardware providing more capacity for less cost. Furthermore, the fast development of computing hardware has enabled the switch from text-based computing to graphical user interfaces and multimedia applications and communications. This transition has fundamentally changed the use of computers and made visual information an inseparable part of everyday computing. Thereby, the lack of effective methods for indexing and retrieving stored information has become the limiting factor for wide utilization of stored visual content.

The traditional text-based approaches to image retrieval have proven out to be inadequate for many purposes. On some occasions, image databases have associated captions or other text describing the image content and these annotations can be used to greatly assist image search. Manually annotating large databases takes, however, a lot of effort and raises the possibility of different interpretations of the image content. As a result, *content-based image retrieval* (CBIR) has received considerable research and commercial interest in the recent years. The field has matured into a distinct research discipline which differs substantially from text-based information retrieval. In CBIR, images are indexed based on the visual content of the image itself, generally using low-level statistical features such as color, texture, and shape. The main advantage is that these features can be automatically derived from the visual content of the images. Visual features are objective, as human involvement is not required in the extraction process, and overall quite natural for visual information processing.

Unfortunately, very few assumptions about image content can be made in the case of general images, and the generic low-level features used in CBIR are insufficient to discriminate this kind of images well on a conceptual level. This creates a quintessential problem in CBIR, namely the *semantic gap* between the high-level semantic concepts used by humans to understand image content and the low-level visual features used by a computer to index the images in a database. Due to the immense need for effective image retrieval applications, a considerable amount of research has been directed on ways to bridge or at least narrow the semantic gap. One common approach is to try to learn the user's preferences with intra-query learning methods such as *relevance feedback*. Relevance feedback is a technique originally proposed for text-based information retrieval to improve the performance of information access systems. The improvement is achieved by modifying the system's responses based on the user's reaction to the previously retrieved items. This way the role of the CBIR system is changed by relevance feedback from an automatic answering machine to an interactive tool that is being used by a skillful human expert. Image retrieval becomes an iterative process of human–computer interaction. Alternatively, relevance feedback can be regarded as a best-of-both-worlds type solution: the retrieval system is an interface between an intelligent high-level system and a low-level system with extremely fast performance on simple, low-level operations. The human user, on the other hand, has a natural ability to explore large amounts of visual data, perform semantic analysis and extract only the relevant information, but is limited in speed and endurance for monotonous tasks.

To answer some of these challenges, the PicSOM project was started in 1998 by Prof. Erkki Oja and Dr. Jorma Laaksonen. It was inspired by the earlier *WEBSOM* project of Academician Teuvo Kohonen and his group, in which Self-Organizing Maps were used for text document indexing and retrieval. In PicSOM, the information is in the form of images instead of text, which sets quite new requirements.

1.1 Contributions of the thesis

The main contributions of this thesis are:

- A survey of various techniques developed and used in the field of CBIR research. The fundamental issues of image indexing, query processing, measuring image similarity, relevance feedback, and retrieval evaluation are addressed.

- The development and representation of our PicSOM CBIR system, including detailed descriptions of using the Self-Organizing Map (SOM) as an image indexing method and a novel relevance feedback technique where multiple parallel SOMs are utilized and user responses are spread to local map neighborhoods. The proposed method provides a common framework for CBIR with the following advantages:

  - It scales well up to large databases of even one million images.
  - It supports the use of multiple features, both visual and non-visual, simultaneously in image retrieval.
  - It has a modular architecture, allowing the easy addition or removal of features.
  - It contains a visualization tool provided by the SOM, which facilitates easy browsing of the database.

- A set of evaluations with different image features (both ones developed by us and ones obtained from external sources), image databases, and parameter settings demonstrating the validity of our approach. A reference technique which ignores the topological ordering provided by the SOM is described and results of comparisons are provided.

- A framework for extending the basis of retrieving relevant images from the visual features. On certain application areas, additional information on the semantic content of the images is available and should be exploited. Especially, a method for using the results of previous search instances or existing keyword annotations as sources of semantic information is provided.

1.2 Outline of the thesis

The following chapters of the thesis are organized as follows. First, a general overview of image retrieval based on the image content is presented in Chapter 2. Chapters 3 and 4 cover the two fundamental phases in CBIR, the image indexing phase and the retrieval phase. First, different techniques for offline processing and indexing the image database are discussed in Chapter 3. Then, an overview of significant issues related to querying and retrieving the indexed content is given in Chapter 4. Relevance feedback is covered in more detail in Chapter 5. Evaluating the performance of CBIR systems is discussed and a summary of performed experiments is presented in Chapter 6. The details of the arrangements and results of these experiments are presented in the included publications. The conclusions of the thesis are drawn together in Chapter 7.

1.3 List of publications and the author's contributions

The journal articles and conference papers listed below are included in this thesis. In this section, the content of each publication is briefly described and the contributions of the author are listed for each publication. The following numbering of the publications is used throughout the thesis when referring to the publications:

I Jorma Laaksonen, Markus Koskela, Sami Laakso, and Erkki Oja (2000). PicSOM – Content-Based Image Retrieval with Self-Organizing Maps, *Pattern Recognition Letters* **21**(13-14): 1199–1207.

II Jorma Laaksonen, Erkki Oja, Markus Koskela, and Sami Brandt (2000). Analyzing Low-Level Visual Features using Content-Based Image Retrieval, *Proceedings of the 7th International Conference on Neural Information Processing (ICONIP 2000)* (invited paper), Vol. 2, Taejon, Korea, pp. 1333–1338.

III Jorma Laaksonen, Markus Koskela, Sami Laakso, and Erkki Oja (2001). Self-Organizing Maps as a Relevance Feedback Technique in Content-Based Image Retrieval, *Pattern Analysis & Applications* **4**(2+3): 140–152.

IV Markus Koskela, Jorma Laaksonen, and Erkki Oja (2001). Comparison of Techniques for Content-Based Image Retrieval, *Proceedings of the 12th Scandinavian Conference on Image Analysis (SCIA 2001)*, Bergen, Norway, pp. 579–586.

V Jorma Laaksonen, Markus Koskela, and Erkki Oja (2002). PicSOM—Self-Organizing Image Retrieval with MPEG-7 Content Descriptions, *IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing* **13**(4): 841–853.

VI Markus Koskela, Jorma Laaksonen, and Erkki Oja (2002). Implementing Relevance Feedback as Convolutions of Local Neighborhoods on Self-Organizing Maps, *Proceedings of the International Conference on Artificial Neural Networks (ICANN 2002)*, Madrid, Spain, pp. 981–986.

VII Markus Koskela and Jorma Laaksonen (2003). Using Long-Term Learning to Improve Efficiency of Content-Based Image Retrieval, *Proceedings of the Third International Workshop on Pattern Recognition in Information Systems (PRIS 2003)*, Angers, France, pp. 72–79.

The PicSOM project has involved a number of people over the five-year period. During the project, four journal articles (Publications I, III, and V and Brandt et al. 2002) and numerous conference papers have been published. In addition, six Master's Theses (Koskela 1999, Brandt 1999, Laakso 2000, Pakkanen 2002, Viitaniemi 2002, Rummukainen 2003) have been written on the subject. The original idea of using multiple parallel Self-Organizing Maps for image retrieval and spreading user responses on local map neighborhoods to achieve relevance feedback was conceived by the manager of the research project, Dr. Laaksonen. The author of this thesis was then hired as the first full-time worker to the project.

Publication I is the first journal article on the PicSOM system. It contains a concise description of all the major parts of the system. The visual features used in early experiments with the system are described. A set of performance evaluation measures used also in later publications and results of retrieval experiments are introduced. The author of this thesis had a substantial role in the work done for the article along with the first author. The experiments were performed by the author. All coauthors participated in the reporting of the work.

Publication II discusses the analysis of low-level statistical visual features in a CBIR setting. CBIR is seen as an emerging research topic benefiting from previous research on natural image statistics. The relevance of different statistical features can be evaluated in the PicSOM setting. While the connection from low-level features to semantic concepts remains unsolved, the use of a meaningful set of parallel features can aid in linking statistical visual features to image similarity perceived by humans. The illustrations of semantic image classes on feature-wise SOM surfaces are introduced. The original features used in and developed for PicSOM are described in some detail. The author of this thesis participated in the general work on this article and implemented some of the visual features.

Publication III is a journal article focusing on the description and qualitative analysis of the relevance feedback technique based on Self-Organizing Maps which is the backbone of the PicSOM system. The article covers also other existing relevance feedback methods and different kinds of usage of SOMs in the CBIR field. Advantages and differences of our method when compared to existing relevance feedback methods are discussed. A general CBIR system structure and the idea of dividing the task of a CBIR system into independent stages or blocks is introduced by separating the per-feature and final processing stages, resulting in reduced computational burden. Initial version of the reference methods based on vector and scalar quantization are introduced. The author of this thesis implemented the reference system, performed the experiments and was a major participant in the preparation of the article.

Publication IV takes the block structure approach further. The operation of a CBIR system is seen as a series of independent processing stages. For each stage, there may exist multiple choices, and different CBIR systems may be implemented in this framework. Moreover, each stage of processing may be analyzed separately. Performed experiments validate these assumptions and show two alternative paths in the block structure that lead to superior results. The author of this thesis participated in the programming work, planned and performed the experiments, and did a major part in the reporting of the work. The first coauthor participated in programming and both coauthors helped in the reporting phase.

Publication V is a journal article describing an extensive evaluation of the PicSOM system with visual content descriptors defined in the MPEG-7 standard. In this work, we have replaced our previous features with ones defined in MPEG-7. In addition, a slightly modified version of our algorithm was presented and used. The results were presented using recall-precision curves and they show that PicSOM can readily utilize the MPEG-7 content descriptors and the system in general benefits from using as many descriptors as there are available without any preceding feature selection. The author of this thesis implemented the modified version of the retrieval algorithm, suggested the use of MPEG-7 descriptors in the PicSOM system, performed the experiments, and participated in the reporting of the work.

Publication VI discusses the interpretation of PicSOM's relevance feedback technique as convolutions of sparse value fields obtained from the relevance information with a kernel function. A number of kernel functions with different sizes are compared in spreading the relevance information on the SOM surfaces. In addition, two methods for incorporating information about the relative distances of the map units in the original feature space are presented. The author of this thesis raised the issue of a comprehensive evaluation of different kernel functions, planned and performed the experiments, and wrote the publication with the help of the coauthors. The first coauthor implemented the location-dependent window functions into PicSOM.

Publication VII presents a method to use previously recorded user–system interaction data as an image feature which can be used to improve retrieval efficiency on large databases of miscellaneous images. The method can also be used for existing keyword annotations, which can result in greatly improved retrieval results. The author of this thesis proposed the method, gathered the user interaction data, performed the experiments, and wrote the publication with the help of the coauthor.

2 CONTENT-BASED IMAGE RETRIEVAL

Image retrieval has been an active research field since the 1970s. The traditional approach has been based on manually inserted annotations describing both the contents of the image and other textual or numeric metadata such as the file name, image format, size, dimensions, and where, when, and by whom the image was created. The user then formulates textual or numeric queries which are made against these annotations. This approach enables the use of the existing and widely adopted methods developed for standard database management systems also in image retrieval. For reviews of text-based image retrieval, see, for instance, Fidel et al. (1994) or the image database system survey by Tamura and Yokoya (1984).

There are, nevertheless, serious drawbacks with the textual annotation approach. First of all, the most descriptive annotations must usually be entered manually. In order to fully describe the contents of an image, the human annotator would have to provide a description for every object's characteristics and spatial and other relations with other objects in the image. This kind of a comprehensive description of images is usually impossible as images contain too much detail. Rather, people tend to enter annotations only for the most obvious content or for the current task at hand. This approach also quickly becomes impractical as the database grows in size. With huge and dynamic databases, such as ones containing indices of the images available in the World Wide Web, this approach is clearly not viable. The annotations may even change later: some previously unnoticed attribute may become an important aspect and, consequently, the images would have to be indexed again to keep the database up-to-date. The second problem is the rich and subjective content images generally have. Therefore, even if the annotations are scrupulously provided for each image in the database, there is a problem with different interpretations of image content different people are bound to have. If the database is large, the annotation task has to be divided among a group of indexers and the interpretations of the images may vary. The user must know the exact terms the annotator used in order to be able to retrieve the images she wants. Usually, this is not the case as the user does not necessarily have any insight into the database generation process. Textual annotations are also language-dependent.

In the early 1990s, content-based image retrieval (CBIR) emerged as a method to overcome the evident problems of text-based image retrieval (see e.g. Kato 1992, Chang and Hsu 1992, Niblack et al. 1993, Gudivada and Raghavan 1995). In the CBIR approach, the images are indexed by features directly derived from their visual content using automatic or semi-automatic image processing techniques. Indexing images differs substantially from indexing textual documents since images or visual information in general do not consist of such fundamental building blocks as words in text which could be directly utilized. Instead, the desired attributes of images for efficient indexing are complex functions of image regions or the whole image. In this sense, image retrieval can be considered as a discipline in the intersection of (traditional) information retrieval (IR) and image processing.

CBIR has received considerable research interest in the last decade and has evolved and matured into a distinct research field. Still, the research field is rather young as the first influential papers were published and the first notable CBIR systems, such as *QBIC* (Niblack et al. 1993, Flickner et al. 1995), *Photobook* (Pentland et al. 1994), *Chabot* (Ogle and Stonebraker 1995), *Virage* (Bach et al. 1996), *VisualSEEk* (Smith and Chang 1996), *MARS* (Huang et al. 1996), and *PicHunter* (Cox et al. 1996, Cox et al. 2000) were developed in the early 1990s. The majority of papers on CBIR have been published after 1995 with a clear upsurge in the last few years.

The interest in the field is a result of both the rapid development of computer hardware and the fact that the need for effective visual information management technologies is immediate. Content in any form has value only if it can be found and the easier it gets to produce visual content, the more complex the problem of managing the content archives gets. Potential applications for image database technologies can be found in diverse fields such as education (Chang et al. 1998b), industry (Iivarinen and Pakkanen 2002), online shopping catalogs (Viitaniemi and Laaksonen 2002), museums and art galleries (Addis et al. 2003), medical imaging (Shyu et al. 1999), geography and remote sensing (Smith 1996), astronomy (Csillaghy et al. 2000), crime prevention and investigation (Pastra et al. 2003), and archiving personal digital photographs and scanned images (Rodden and Wood 2003), among many others.

In recent years, textbooks on CBIR and multimedia retrieval in general have begun to appear, including Gong (1998), Del Bimbo (1999), Lew (2001), Santini (2001), Castelli and Bergman (2002), and Dunckley (2003). Several worthy survey articles have also been written, including Gudivada and Raghavan (1995), Aigrain et al. (1996), Forsyth et al. (1996), Chang et al. (1997), Gupta and Jain (1997), Jain (1997), Eakins and Graham (1999), Rui et al. (1999), Yoshitaka and Ichikawa (1999), Smeulders et al. (2000), Vasconcelos and Kunt (2001), Eakins (2002), and Antani et al. (2002). In addition, surveys focusing on reviewing CBIR systems have also been compiled (Johansson 2000, Venters and Cooper 2000, Veltkamp and Tanase 2000).

2.1 Types of image search tasks

General CBIR systems must support a multitude of usage types. Users of a CBIR system are likely to present a very diverse set of different search scenarios, which the system should support. A commonly used classification for CBIR search tasks is given in Cox et al. (2000). The most precise search task is *target search*, in which the user is trying to find a specific target image which may or may not be actually present in the database and which is the only relevant image for this query. An example situation for a content-based target search takes place when the user is searching for an image of a previously seen painting, knowing neither the name of the artist nor the title of the painting. Generally, target search is mostly employed when searching for a specific known image in catalogs, component listings, trademark or logo databases, personal photograph collections, etc.

Category search occurs when the user is looking for images belonging to a certain category or class of images and all images fulfilling the category criteria are considered relevant. The remaining images are then non-relevant to the query. Here, the notion of a class should be considered a user-centric concept, used as an aid for this discussion, rather than a "hard" class as is generally assumed e.g. in pattern recognition. In the beginning of a category search, the user may have initial example images belonging to the class in question to start the search with or the search may be initiated using some alternative method such as a keyword query or a preceding image browsing phase. In CBIR system performance evaluations, a common test setup is to assume a single initial example image and that the user is looking for additional images in the same class. Category searches may be enhanced during the query in a natural way by relevance feedback, i.e. grading the returned images on whether they belong to the class in question and communicating this information back to the retrieval system, thereby providing more information about the class of relevant images and thus guiding the system toward the remaining relevant images in the database. An implicit assumption of both target and category searches is that the user is able to partition the image database $\mathcal{D}$ into sets of relevant and non-relevant images, $\mathcal{D}^{\oplus}$ and $\mathcal{D}^{\ominus}$, respectively, and that $\mathcal{D} = \mathcal{D}^{\oplus} \cup \mathcal{D}^{\ominus}$ and $\mathcal{D}^{\oplus} \cap \mathcal{D}^{\ominus} = \emptyset$. In target search, the set $\mathcal{D}^{\oplus}$ contains only one image, $\mathcal{D}^{\oplus} = \{\mathcal{I}^{\oplus}\}$. The validity of this assumption has been questioned in IR (see e.g. Ingwersen 1992) and many researchers consider relevance to be a fuzzier and more pragmatic concept. The evaluation of IR and CBIR methods becomes, however, much more complicated without making this assumption.

In *open-ended search* or *browsing*, the user has a vague or inexact search goal in mind and she browses the database for any interesting things. The retrieval goal can abruptly change during the session when the system returns interesting but unexpected images. Image searches of this type are highly interactive and often constitute a nonlinear sequence of actions, thus requiring a flexible user interface. A database visualization tool providing an overview of the database as well as a localized point-of-interest with increased level of detail is needed. In addition, relevance feedback is a useful way to manipulate the system toward the desired kind of images also in open-ended searches. It should be noted that objective evaluations of system performance become increasingly difficult as we move from target search to less exact search tasks.

The above classification is a useful starting point but certainly does not cover all image retrieval tasks. Another way to examine the issue is to take a more user-centric approach by studying actual users and their retrieval practices. For example, the image retrieval needs of art directors have been studied by Garber and Grunes (1992) and of journalists by Ornager (1997) and by Markkula and Sormunen (2000). User studies comparing the traditional keyword-based approach with a retrieval method based on semi-automatic spatial indexing and metadata-based categorization of the images have been presented by Jose et al. (1998) and Yee et al. (2003), respectively.

2.2 Semantic gap

Depending on the image domain, the type of the image query, and the amount of *a priori* information available on the images, the CBIR problem exhibits a varying degree of difficulty. The fundamental problem is that, for a computer, extracting the semantic content from an image is an exceedingly difficult task as objects with the same semantic content often have variable visual appearances and many semantically totally different objects are visually nearly similar (Gupta et al. 1997). In particular, automatic segmentation and object recognition in general images are very difficult problems and even if these were completely solved, it would not be enough for defining image semantics in the general case (Santini et al. 2001). Humans, on the other hand, possess a highly sophisticated visual system and have a lot of *a priori* information on different objects, which we automatically use e.g. in object recognition. This information is based on previous experience, personal preferences and interests, cultural issues, and the context in which the image is represented. Unfortunately, this kind of knowledge is inherently hard to duplicate in a computer vision application. This discrepancy is commonly referred to as the *semantic gap*. Feature extraction methods developed in computer vision, image processing, and more recently also directly in the CBIR field can be seen to provide different solutions to this problem. The digital image itself, consisting of a regular array of pixels with different color or gray-level values, is clearly a representation not suited for semantic analysis. In this sense, one can consider the task of feature extraction, that is, finding as good features as possible, as a step in bridging the gap between raw image data and image semantics.

A straightforward and influential factor in the complexity of the image retrieval problem is the repertoire of images in the database—the image domain (Smeulders et al. 2000). A *narrow image domain* has only a limited and predictable variability in all aspects of appearance whereas a *broad image domain* has unlimited and unpredictable variability as well as ambiguous and subjective semantics. The scope of a given image domain is largely a subjective issue lacking an exact definition. However, an effort to develop a measure of image database complexity, analogous to the concept of *perplexity* of a text corpus, was presented in Rao et al. (2002).

A common narrow-domain test set, also for CBIR research, is the Brodatz texture collection which provides images with somewhat homogeneous stochastic textures (Brodatz 1966). Object databases, in which sets of physical objects have been photographed in a controlled setting with a uniform background, are also typical narrow-domain CBIR test collections. An example of such is the Columbia Object Image Library (COIL-100) (Nene et al. 1996). Of real-world application areas involving narrow image domains, the most studied one is undoubtedly retrieval of trademark images, typically based on shape features as the lack of background enables automatic segmentation of the images, see e.g. Eakins et al. (1998), Jain and Vailaya (1998), Ciocca and Schettini (2001), King and Jin (2001), Yin and Yeh (2002), or Neumann et al. (2002). Other narrow domains include, among many others, different kinds of medical images (Shyu et al. 1999), face recognition (Pentland et al. 1994), maps (Samet and Soffer 1996) and industrial applications such as paper web defect images (Iivarinen and Pakkanen 2002). The results of applying CBIR in these domains have been rather good, as is to be expected.

At the other end lie broad image domains, often containing large quantities of unconstrained images with little general or domain-specific information available. The semantic content of the images is variably unrestricted and heterogeneous. General photograph and other stock image collections constitute typical broad domain databases. An important image domain of this type is the World Wide Web which has, in addition to the enormous database size, its unique challenges due to its dynamic and unlocalized nature. Since very few assumptions about the images can be made, only representations of very general nature are valid. Object recognition and image understanding, even in a limited sense, are generally impossible. The performance of automatically extracted visual features remains moderate and additional, e.g. non-visual or semi-automatic, features may well be required for reaching an acceptable retrieval performance level.

The varying difficulty of the CBIR problem can also be examined from the viewpoint of different users' needs. Generally, users are interested in searching for images of particular semantic attributes, scenes, objects or events, rather than based on low-level similarity in visual content. In fact, the query the user has in mind may be so abstract that the user herself does not know or is unable to explain what she is looking for until she finds it. Image features were explicitly divided into primitive or low-level features and logical features denoting deeper semantics manifested in the images by Gudivada and Raghavan (1995). A similar categorization of image retrieval was proposed by Eakins (2002) who identified three distinct levels of image queries:

- Level 1, retrieval by *primitive (visual)* features.
- Level 2, retrieval by *logical* features or *semantic* attributes.
- Level 3, retrieval by *abstract* attributes.

This framework emphasizes the mismatch between user needs and capabilities of current CBIR systems. Level 1 queries concentrate on basic low-level components of visual content. Used features are typically based on color, texture, shape, and spatial arrangement of uniform regions in the image. Level 2 introduces semantics to the queries. Queries at level 2 may contain specific objects (e.g. “car”) and scenes (e.g. “beach”). At this level, some degree of object and scene recognition as well as inference about the image content is required. At the highest level of complexity, operation at level 3 involves sophisticated image understanding, knowledge representation, and reasoning about the relations and significance of objects and scenes, which goes beyond the enumeration of objects and their relations in the image. Level 3 queries may contain abstract concepts (e.g. images depicting “freedom” or “humor”). Users formulate queries mostly on levels 2 and 3 and expect the systems to operate at the same levels of complexity and semantics but the current CBIR systems operate mainly at level 1. Fortunately, there is substantial overlap between levels 1 and 2 in many cases, which makes it possible to develop CBIR systems with sufficient performance for many applications. In other words, it can be stated that the underlying assumption in CBIR is that semantically similar images also share similar visual characteristics that can be automatically extracted or that semantic features can be synthesized from the low-level features with automatic techniques. The limitations of current technologies for image processing and understanding restrict, however, the validity of this assumption to hold only to a certain level. On the other hand, this overlap may also cause frustration on inexperienced users as a CBIR system may at first appear to operate genuinely on a higher level, but further retrieval results may be disappointing from this viewpoint.

In the research field, the deficiencies of current image retrieval techniques have been long noted and recent research has been increasingly focusing on moving toward level 2 retrieval. In this research, the leading principle is to build semantic representations by extracting intermediate semantic levels from the low-level features (see e.g. Chang et al. 1998a, Naphade et al. 1998, Colombo et al. 1999). For success, relatively moderate objectives must be placed and therefore these techniques are often dubbed as finding *weak* or *simple semantics*. Recent reviews on semantic image retrieval include Eakins (2002) and Naphade and Huang (2002). Level 1 multimedia processing is, however, facing a daunting task with level 3 queries and a fundamental paradigm shift may be required for real semantic retrieval. Indexing and retrieval at level 3 is currently possible only by using textual descriptions.

The weakness of the connection between semantic concepts and visual low-level features is a serious limitation and reduces the usefulness of the content-based approach. As a result, many content-based retrieval applications cannot be expected to produce the best available images as the first response or reach high precision of relevant items. They can, nonetheless, serve as valuable semi-automatic tools and make retrieving images manageable even from large-scale general image collections. Satisfactory results can often be obtained if the image query can be turned into an iterative process toward the desired retrieval target. In this setting, the focus is shifted from formulating elaborate one-shot queries to match only to the relevant items as well as possible, into progressive interaction between the user and the retrieval system. This kind of operation is commonly denoted as relevance feedback and it will be the topic of Chapter 5. In many applications, a relatively low precision can be acceptable, provided that the system is ultimately capable of returning the correct image with a reasonable effort. It should be highlighted that, in this sense, CBIR is intrinsically different from many traditional pattern recognition problems, in which the different classes are much more easily defined and separable and, therefore, low probabilities of error can be achieved. In fact, in the current state of machine vision technology, the task of a CBIR system in semantic retrieval should be mainly seen as reducing the number of returned images in an image query compared to random browsing or systematic examination of the database.

3 FEATURE EXTRACTION AND INDEXING

Modern databases, and ones consisting of images or other visual data are not an exception, are regularly used to store large amounts of data. Database operations are thus typically data intensive as opposed to many common computing tasks, especially in scientific computing, which are computationally intensive. At the simplest form, an image database is only a collection of images. Computational requirements for supporting effective browsing and retrieval, however, demand that we use some kind of an organized structure and means for rapid access to the database. Therefore, among the first and most crucial tasks in constructing an image database application is to compile a suitable *index* to the database. In the scope of this thesis, an index can be defined as any data structure over the original data which enables efficient retrieval. Review articles focusing on different aspects of indexing visual and multimedia data include Idris and Panchanathan (1997), De Marsicoi et al. (1999), Böhm et al. (2001), and Lu (2002).

Image indexing and the preceding feature extraction are typically performed offline, during the construction and setup of the database application. Even dynamic databases, in which images are added and removed during operation, are often reorganized and reindexed in background with offline-type processing. This may be performed at regular intervals or when a sufficient number of changes (insertions and deletions) have occurred. Consequently, computational requirements for generating the index are usually not as crucial as for tasks in the query stage. Instead, the attention is focused on optimizing the system for search tasks i.e. minimizing computations needed during online operation. Calculations should therefore be performed in advance as much as possible and the results stored for later utilization. In some cases, certain indexing methods are also required during online operation, mainly if the system supports adding new images straight to the database or if external images can be used as the starting-point for image queries.

This chapter deals with the two main parts of offline processing required for a content-based image retrieval system: extracting suitable features to describe the images and constructing efficient indices for the features. First, we proceed with a summary of different feature extraction methods commonly in use in current retrieval systems. Next, an overview of common techniques for indexing the extracted feature representations is presented. A more detailed description of the indexing method used in our work, the (Tree Structured) Self-Organizing Map, then ends this chapter.

3.1 Feature extraction

The construction of a CBIR index begins with the extraction of suitable *features* from the images in the database. A feature refers to any characteristic which, in some way, describes the content of an image. In a broad sense, this includes visual features extracted directly from the raw image data, textual keywords, captions, and annotations, and also other kinds of textual or numeric metadata associated with the image. In feature extraction, each image in the database is transformed with M sets of different feature extraction methods to a set of M low-dimensional prototype vectors in the respective feature spaces. The mth representation of the ith image $\mathcal{I}_i$ is compiled into a K_m-dimensional feature vector $\mathbf{f}_i^m = (f_i^m(1)\ f_i^m(2)\ f_i^m(3)\ \ldots\ f_i^m(K_m))^T$. This kind of *vector space model* (VSM) representations with fixed dimensionalities K_m for the features is generally assumed in this thesis. Typical values for K_m in content-based image retrieval are of order 100 (Rui et al. 1999).

Two main categories of image features, *primitive* and *logical* features, were identified in Gudivada and Raghavan (1995). We follow this categorization here, but refer to these basic types as *visual* and *semantic* features, respectively. Visual feature extraction is the foundation for all kinds of CBIR applications and, therefore, various types of visual features have been developed and studied. In fact, most of the early work in the field was concentrated on finding the best possible features to represent different kinds of images to facilitate effective retrieval. The conventional requirement that the features can be automatically extracted, however, limits features of this type to low-level statistical representations. Semantic features, on the other hand, are abstract representations of images manifesting deeper semantics at various levels of detail and describing objects, scenes, events, and also abstract content within the image. In general, semantic features cannot currently be obtained for unconstrained images without human involvement at some stage of the extraction process. These two basic types of features will be discussed in more detail below.

3.1.1 Visual features

The simplest visual image features are directly based on the pixel values of the image. This kind of features are, however, very sensitive to noise and varying imaging conditions and not invariant e.g. to affine transformations. Using image pixels directly is also very inefficient. For example, should we use the pixel values of an $n \times m$-sized gray-level image as features, the image would be transformed into a feature vector in an nm-dimensional space; for a color image the dimensionality would be threefold. Processing large numbers of such vectors is clearly infeasible due to massive storage and computation requirements. As a result, direct pixel-based features are seldom used in practice. Instead, more practical visual features can be obtained by computing certain characteristics or signatures from the images by using suitable image processing or computer vision techniques. This way, the original dimensionality of the image data is reduced during the feature extraction process and, as in dimensionality reduction in general, a good feature maintains those characteristics of the original data which preserve the discriminating power while excluding any redundant information. In general, low-level visual features can be either statistical or structural (syntactic) in nature. However, with images whose content is unrestricted, generally only statistical features can be called upon since the structural approach requires a definite structure which can be captured with a composition of derived rules.

Visual features can be extracted either with automatic or semi-automatic methods. Fully automatic feature extraction is appealing for obvious reasons, especially with large or dynamic databases, but the current level of knowledge on image analysis and pattern recognition techniques is limited and the automatic methods at our disposal cannot always provide sufficient discriminating power for effective image retrieval. Semi-automatic methods, on the other hand, rely on human assistance in tasks like image segmentation. For example, since the recognition of objects in general images is a very difficult task for a computer, manually pointed object contours can be used to enhance shape detection and thus shape-based image indexing. Using semi-automatic methods can lead to notable performance improvements but, depending on the application, the requirement of human effort can be intolerable. As a result, for indexing large collections of miscellaneous images, the repertoire of available features is generally restricted to global features, features computed from fixed image regions or zones (as was done in Publications I–IV or e.g. in Stricker and Dimai (1996) and Taycher et al. (1997)), features relying on *weak segmentation*, i.e. finding internally homogeneous regions according to a specific feature or features instead of actual object recognition (see e.g. Ma and Manjunath 1997, Carson et al. 2002, Barnard et al. 2003, Sjöberg et al. 2003), or to features based on identifying *interest points* in the images, i.e. pixel locations at which the image signal changes two-dimensionally (see e.g. Schmid and Mohr 1997, Bres and Jolion 1999, Loupias and Sebe 2000, Amsaleg and Gros 2001).

Usually, the general-purpose visual features, applicable for a variety of image types, are said to include color, texture, and shape. These feature types have been extensively treated in many review articles as well as in CBIR textbooks; see, for example, the pertinent chapters of Lew (2001) and Castelli and Bergman (2002). MPEG-7, a noteworthy standardization initiative for describing multimedia content (see Section 6.3) also follows this categorization, recognizing color, texture, and shape as the three fundamental types of visual features applicable to automated still image content description. MPEG-7 also defines a set of standard visual features or Descriptors which have also been used in this work (Publications V–VII). Sometimes also the structure or composition of the image is mentioned as a basic feature type, although it severely suffers from the requirement of prior segmentation. Other feature types are generally specific to certain application domains and require special domain knowledge and constrained images. This makes these features ill-suited for general use and therefore fruitless to consider outside the specific application context.

Color. Color is a simple and straightforward feature for all kinds of color images. The human eye is much more sensitive to color shades than gray-level intensities in an image. The colors of different objects are also largely resolution and view invariant. Selecting an appropriate color space and the used color quantization are key issues for color feature extraction. Smith (1997) lists the elemental properties for a feasible color space as uniformity, completeness, compactness, and naturalness. Still, the ordinary *RGB* color space is commonly used, although it suffers from not being perceptually uniform. Perceptually more uniform color spaces, such as *HSV* and *L*a*b** (Jain 1989), can be obtained from *RGB* by using a nonlinear transform. Color quantization is used to reduce the number of distinct colors in an image. It is used to reduce both computational complexity of color feature extraction and the dimensionality of the resulting feature vectors.

Color has been the most commonly-used feature type in CBIR. Basic color features are easy to implement and usually yield reasonable and predictable results which can then be improved by including other types of features. The standard representation for color information in CBIR has been the color histogram, first investigated in this context by Swain and Ballard (1991). The color histogram describes the distribution of different colors in an image in a simple and computationally efficient manner. Other commonly used color features include color moments (Stricker and Orengo 1995), color regions (Hsu et al. 1995), color sets (Smith and Chang 1995), the color coherence vector (Pass et al. 1996), and the color correlogram and autocorrelogram (Huang et al. 1997b).

Texture. Texture is an innate property of all surfaces referring to visual patterns not resulting from the presence of a single color or intensity. Albeit being intuitively obvious, texture lacks a precise definition. Humans often distinguish textures with properties like periodicity, directionality, granularity, and randomness. Because of the importance and usefulness of texture information, various texture representations for diverse application areas in pattern recognition and computer vision have been extensively researched over the last decades and these achievements are now being adapted also to CBIR applications. Generally, an image can be considered to be composed of a number of salient regions with different texture patterns and the properties of these regions can be used in image indexing.

Texture analysis methods can be divided into syntactic and statistical approaches. In syntactic texture analysis, different textures are described with suitable and distinct grammars by setting single pixels or connected sets of similar pixels as primitives and defining their allowed spatial arrangements. Syntactic methods work best with deterministic or "strong" textures having large and distinct primitives. Statistical methods, on the other hand, describe textures according to their underlying statistical properties. Each texture is described by a feature vector. Various statistical methods have been studied and used in texture analysis as they are more suitable for describing many stochastic or "weak" real-world textures. Thereby, statistical methods are dominant in CBIR.

Texture representations readily applicable to CBIR include the co-occurrence matrix (Haralick et al. 1973), Tamura representation (Tamura et al. 1978), SAR/MRSAR texture models (Mao and Jain 1992), Wold decomposition (Liu and Picard 1996), Gabor functions (Turner 1986), wavelets (Daubechies 1990), and local binary patterns (Ojala et al. 1996) among many others.

Shape. Shape features have not been studied as intensively as color and texture from the CBIR viewpoint. This is mostly due to inherent difficulties in object recognition and shape representation and the lack of a mathematical formulation corresponding to the human perception of shapes. Shape has, however, the potential to be the most important representation in many application areas.

In order to enable querying for specific objects, the system would need a shape description capable of distinguishing different shapes and regions belonging to separate objects. The shape of an object is, unfortunately, very much dependent on the view and distance. A 3D object can be projected into a 2D image in a variety of ways. As a result, the task of general object recognition is beyond current technologies. Still, shape is an important source of information for CBIR, especially in restricted image domains where robust segmentation is possible. Shape-related global features, i.e. ones which operate on the whole image and therefore do not require object detection have also been developed for CBIR on general images. Within our research project, a study of statistical shape features not requiring segmentation was presented by Brandt et al. (2002).

Shape representations can be divided into two general categories: boundary-based and region-based methods. Boundary-based methods utilize only information about the boundary of an object, whereas region-based methods describe shapes based on the whole area of the object. Thus, the intrinsic difference between these representations is that boundary-based methods model the object as a one-dimensional curve while region-based methods operate on two-dimensional fields. Common boundary-based shape features applied to CBIR include chain codes (Freeman 1974), Fourier descriptors (Zahn and Roskies 1972, Persoon and Fu 1977), and Wavelet descriptors (Chuang and Kuo 1996). Moment invariants (Hu 1962), Zernike moments (Khotanzad and Hong 1990) and simple heuristic region features, such as area, Euler's number, circularity, eccentricity, elongatedness, and rectangularity, are common examples of region-based shape features in CBIR applications.

3.1.2 Semantic features

In order to make genuine image indexing by higher-level content possible, an inevitable requirement is to be able to capture the image's semantic content in such a way that it corresponds to the human view of image semantics. As was discussed in Section 2.2, with general images, automatically extracted visual features often fail to do this adequately and additional sources of information are needed for reaching acceptable performance. Naturally, automatic extraction of semantic content would be a decisive step forward for CBIR and related fields. For example, in Naphade and Huang (2002), it was dubbed "the final frontier" of multimedia indexing. In some cases, however, certain semantic categorizations are possible with current automatic methods. Types of semantic image categories can be distinguished with specialized classifiers which typically perform two-class classifications to the database images. This kind of semantic image categorization can be seen as a very limited form of image understanding where the task is to assign one or more semantic classes to each image, instead of trying to comprehensively understand image content. Experimental methods for this type of image categorization have been developed, for example, to distinguish photographs from computer-generated images (Frankel et al. 1996), indoor and outdoor images (Szummer and Picard 1998), and city images from landscape scenes (Vailaya et al. 1998). These single two-class classifiers can then be combined to achieve more extensive categorizations. For example, hierarchies of classifiers can be constructed as in Vailaya et al. (2001), where the resulting categories from earlier classifiers are further classified to more specific categories. Another example of a CBIR system using semantic categorization methods is presented in Wang et al. (2001). Automatic annotation of images in a broader setting has also been studied, and some results are presented in Barnard and Forsyth (2001).

In some occasions, the image database or a portion of it may already contain elaborate manually-constructed captions or other annotations. Such annotated databases can typically be found e.g. in commercial image libraries, art galleries, news photo archives, and medical image databases. For example, the Corel Photo CDs widely used in CBIR research contain keyword annotations. A method for using these keywords as an image feature is presented in Publication VII. Implicit annotations can also be found, e.g. from the text surrounding an image in the WWW. In fact, due to the immense popularity growth of the WWW, combining text from associated HTML pages and image features to enable semantic image retrieval from the WWW has become a widely studied issue (see e.g. Agnew et al. 1997, Sclaroff et al. 1999, Lew 2000, Aslandogan and Yu 2000, Newsam et al. 2001, Zhao and Grotsky 2002). A straightforward way to use these annotations is to implement text-based retrieval, in which case the problem transforms into one of traditional information retrieval (IR). The annotations are used as a textual document associated with the image and the best-matching items to a query are determined using standard IR techniques but, instead of the associated text, the corresponding images are returned as the result of the query. The problem of possible keyword mismatches between the query and the image annotations can be alleviated by using a general-purpose electronic thesaurus such as *WordNet* (Fellbaum 1998) (examples of using *WordNet* in image retrieval include Aslandogan et al. (1997), Duffing and Smaïl (2000), Benitez and Chang (2002), and Han and Guo (2002)) or by generating an automatic thesaurus from the annotated image database (Zhou and Huang 2002). An alternative method is to use the annotations indirectly, as sets of binary attributes affixed to the images. Each keyword or term in the annotations is represented as a binary attribute and images with that term in their annotations have the corresponding attribute set to one. These *hidden annotations* (Cox et al. 1997) can then be used like any other statistical feature to represent images in the database. This approach may be useful if the annotations correspond to complex semantic similarities that are not easily explained or if the vocabulary or the language of the annotations is unknown to the user.

One avenue of research in semantic image retrieval has been to study methods for reducing the workload needed for image annotation and provide tools to aid the annotation process in a semi-automatic manner. The goal in these methods is to require manual annotations only for a small fraction of images and to use automatic methods to assign probable annotations to the remaining images. Active learning has been used for this task, so that, during the learning stage, the system prompts images for manual annotation based on how much the annotations can decrease the uncertainty in the system (Zhang and Chen 2002, Sychay et al. 2002). Interactive tools for image annotation have also been developed e.g. by Minka and Picard (1997), Srihari and Zhang (2000), Schreiber et al. (2001), and Pfund and Marchand-Maillet (2002).

Previous user interaction with the CBIR system can also be recorded and used to infer information about the semantic content of the images. During a query, the user implicitly evaluates images according to her current information need. The fact that two images are given similar relevance evaluations during a single query session is a cue for similarities in their semantic content. Extracting these features are discussed in more detail in Section 5.5 and Publication VII.

Semantic information can also be available on application-specific sources depending on the application area. For example, the hypertext link structure of the WWW can be used to construct a statistical image feature as was done in Laakso et al. (2001). The basis of the method consists of a set of basic relations that can take place between two images in the WWW. For example, if one image acts as a hypertext link to another image (e.g., as a thumbnail) it can be assumed that the two images are closely related. Also, if two images are situated on the same WWW page, it is likely that they are somehow semantically related. Furthermore, a noteworthy initiative in the WWW domain for extending the current WWW with semantics is the W3C's Semantic Web effort (Berners-Lee et al. 2001, W3C 2003). The aim of the Semantic Web is to provide a framework for improving the cooperation of computers and people in the WWW based on the Resource Description Framework (RDF) language.

3.2 Indexing techniques

Indexing multimedia databases is a different and in many ways more complex problem than indexing traditional databases. The main difficulties arise from the high dimensionality K of the typically used feature vectors. High-dimensional spaces lack many intuitive geometric properties we are accustomed to in low-dimensional spaces (Castelli and Bergman 2002). We cannot properly imagine high-dimensional spaces so we try to find low-dimensional analogies where the same effects may not occur. These difficulties are commonly subsumed into the term *curse of dimensionality* (Bellman 1961). In addition, the size of the image database can be large and it may be required to rely on using many features simultaneously in image retrieval (this will be discussed in Section 4.2.2). Due to these factors, using basic linear search, where every stored feature vector is considered, easily leads to poor performance. Fast response time is, however, essential in interactive systems as users are quick to reject systems they consider overly sluggish (see e.g. Nielsen 1994). Therefore, specialized techniques and efficient data structures are needed to manage the retrieval process so that the best-matching images can be determined quickly enough.

A typical task in image retrieval is to determine k nearest data items to a specific point in a high-dimensional feature space, denoted as *k-nearest-neighbor* (kNN) query. Other types of queries (i.e. point, range, and within-distance queries, see Section 4.2) are not as important in image retrieval, so the focus in this discussion is on index structures supporting kNN queries. The concept of a nearest neighbor requires a similarity or distance measure, which will be discussed in Section 4.2.1. In general, there are two broad categories of index structures for high-dimensional spaces. The first approach is to apply a divide-and-conquer strategy. The data or the feature space is divided into categories (clusters) or subspaces with the intention that only one or a few of these have to be processed in one given query. Alternatively, we can transform the original feature space into a new space where the operations needed to process a database item are less demanding. This usually means reducing the dimensionality of the original feature space.

A number of common indexing techniques for high-dimensional features are briefly discussed next. The list is not comprehensive due to the vast number of different techniques that have been presented over the years of research in the field. Instead, the intent of this presentation is to provide a concise overview of different methods and to emphasize parts of research most related to the present work. For more detailed treatments of the subject, an interested reader is directed to the reviews on indexing listed in the beginning of this chapter as well as the general works on CBIR listed in Chapter 2.

3.2.1 Dimensionality reduction

The distribution of image feature vectors in high-dimensional spaces is typically not uniform, but rather has local structure. Also, the features represented at the feature space spanned by the dimensions are often highly correlated, i.e. the intrinsic dimensionality of the data is lower than K. These properties make it feasible to approximate the original space by projecting it into a new space with a lower dimensionality and thus reduced computational requirements. Still, this inevitably results in a loss of information. For kNN queries, the loss of local proximity information between data items is most harmful and should be minimized.

The mapping from a higher-dimensional to a lower-dimensional space, i.e. dimensionality reduction, can be accomplished with linear methods like variable subset selection, principal component analysis (PCA) (Hotelling 1933), singular value decomposition (SVD), random projection (Kaski 1998) or nonlinear methods such as multidimensional scaling (MDS) (Kruskal 1964) or Self-Organizing Map (SOM) (see Section 3.3.1). Examples of using dimensionality reduction methods for image indexing and retrieval, in addition to our SOM-based method discussed in Sections 3.3 and 5.4, include Beatty and Manjunath (1997), Ravi Kanth et al. (1999), Kulkarni et al. (1999), and Wu et al. (2000a). A more recent method is independent component analysis (ICA) (Comon 1994, Hyvärinen et al. 2001) which has been applied also in image retrieval (Kolenda et al. 2002). Since dimensionality reduction reduces the complexity of measuring similarity between data items, it may be sufficient on its own to facilitate effective retrieval. An alternative approach is to use dimensionality reduction as a preprocessing step and then some indexing method on the lower-dimensional space.

3.2.2 Recursive partitioning methods

Recursive partitioning methods divide the feature space or the data set into progressively smaller partitions. The resulting hierarchical structure is then represented as a tree and the efficiency of accessing data items is significantly improved by utilizing the hierarchy. Existing methods differ in the way the partitioning is performed. Castelli and Bergman (2002) listed quadtrees (Finkel and Bentley 1974), k-dimensional trees (k-d-trees) (Bentley 1975), and R-trees (Guttman 1984) as the most commonly used families of recursive partitioning methods. Quadtrees divide a K-dimensional space into 2^K regions by splitting the space into two parts in every dimension. Each node of the tree is thus either a leaf or has 2^K immediate children. The k-d-tree is a k-dimensional extension of the standard binary tree. It divides the space by using $(K-1)$-dimensional hyperplanes one dimension at a time. R-trees apply possibly overlapping hyperrectangles, represented as nodes in the tree, to divide the space. Children of a node then further divide the space inside the hyperrectangle with smaller hyperrectangles.

Originally, the above methods were developed for lower-dimensional spaces and point or range queries (Section 4.2), so extensions to these basic methods are required for using them effectively in multimedia indexing. The main problem is that these methods do not scale well with regard to dimensionality. Therefore, they are mostly useful for medium-dimensional ($K < 20$) feature spaces. According to a study by Weber et al. (1998), under certain assumptions, a simple linear search outperforms these methods already when the dimensionality exceeds $K = 10$. Secondly, the performance of a kNN query often suffers if the query point is located near a partition border; either we take also the neighboring partitions into account, resulting in increased computational requirements, or risk degrading retrieval precision as a portion of potential images are ignored. Insightful reviews and comparisons of various recursive partitioning methods are presented by White and Jain (1996a) and Böhm et al. (2001). As a rule of thumb, it can be stated that, regardless of the used partitioning method, the search time of kNN queries in medium and high-dimensional spaces increases exponentially with dimension and linearly with the number of nearest neighbors. Recommended indexing methods for kNN queries in various sources include optimized versions of the k-d-tree (see e.g. Egas et al. 1999, Castelli and Bergman 2002), R*-tree (Kriegel et al. 1990), X-tree (Berchtold et al. 1996), SS-tree (White and Jain 1996b), VA-file (Weber et al. 1998), and Pyramid-tree (Berchtold et al. 1998).

3.2.3 Clustering

Clustering means partitioning data into m sets or clusters so that data items in a certain cluster are more similar to each other than to data items in other clusters. In the basic form (also called hard or crisp clustering), every data item belongs to exactly one cluster. Clustering can be used to produce an effective image index as follows. After clustering, each cluster is represented by its centroid or sometimes a single representative data item (i.e. the *image label* for that cluster) and, instead of the original data items, the query point is compared to the centroids or the cluster representatives. The best cluster or clusters, according to the used similarity measure, are then selected and the data items belonging to those clusters are evaluated and k nearest neighbors are returned. If the number of clusters is high, we can further cluster the centroids to obtain clusters of clusters, i.e. superclusters, or use some hierarchical clustering method in which the data is gradually clustered from the original data to a single cluster. Many clustering methods have been proposed for image indexing, including competitive learning (King and Lau 1997), the ClusterTree algorithm (Yu and Zhang 2000), agglomerative hierarchical clustering (Duffing and Smaïl 2000), vector quantization (k-means clustering) (Publication IV or V, or e.g. Chen et al. 1997, Wood et al. 1998, Iyengar and Lippman 1998, Lu and Teng 1999, Yoo et al. 2002, Qiu 2002, Ye and Xu 2003), k-medians clustering (Volmer 2002), and SOM (see Section 3.3.1; or Vesanto and Alhoniemi (2000) for a general study on using SOM for clustering).

3.2.4 Vantage points

Vantage point methods rely on selecting a set of m vantage points (a.k.a. interest points or anchors) for which the similarity to all data items is calculated during the indexing phase. This way we get an ordering of decreasing similarity for the data items to each of the m vantage points. Images with similar feature vectors are located in similar positions in these orderings due to the triangle inequality. Clearly, images with dissimilar feature vectors may also end up in nearby positions but their count can be reduced by using multiple vantage points. Still, false positive findings can remain but the indexing method guarantees zero false negatives. During query time, we can thus obtain all similar data items to the query by computing the similarity of the query point to the vantage points and selecting all images that have alike similarity values to all vantage points. False positives can then be eliminated by calculating the similarities of the candidates in the original feature space.

Indexing by vantage points has similarities with both dimensionality reduction and clustering. By using m vantage points, each data item is represented as a point in a new m-dimensional space through a nonlinear transform. In clustering, cluster centroids share a similar purpose as vantage points. The difference here is that only the nearest centroid matters in clustering whereas these methods store and use the similarities from data items to all vantage points. Studies on using vantage points in image indexing include Vleugels and Veltkamp (2002) and Natsev and Smith (2002).

3.2.5 Inverted file

In text-based IR, individual documents in a corpus are often represented by the words they contain and all structure is neglected i.e. the so called "bag of words" model is assumed. The contents of the documents are gathered into a term-by-document matrix $\mathbf{X}$ where the (j,k)th element of $\mathbf{X}$ is a function of the number of times term j occurs in document k. Typically, any given document contains only a small subset of the available terms and certain terms occur very frequently. This phenomenon is commonly referred to as the Zipf's Law: $f \propto 1/n$, i.e. the frequency f of a word is inversely proportional to its frequency rank n. This means that $\mathbf{X}$ is typically very sparse and the similarity computation can be restricted to a small subspace spanned by the query terms. This enables us to use an efficient indexing technique called the *inverted file*, which contains an entry for each possible term with a list of documents containing that term.

Inverted files can also be used in image retrieval, provided that the image features fulfill the requirement of sparsity. Generally, this is not the case but the features can be especially designed so that this condition is fulfilled. A well-known example of using inverted files in image retrieval is the *GIFT* system (Squire et al. 1999a, Squire et al. 2000).

3.3 Tree Structured Self-Organizing Maps

The main image indexing tool used throughout the work constituting this thesis and in the PicSOM system is the Self-Organizing Map (SOM) (Kohonen 1982, Kohonen 2001). The object has been to utilize the strong self-organizing power of the SOM in unsupervised statistical data analysis for image retrieval.

In this section, the use of the SOM as an image indexing method is discussed. First, we review the standard SOM and discuss its usage in this application field. The SOM is then augmented with a hierarchy using a tree structure, which provides useful properties for the algorithm especially in image retrieval.

3.3.1 The Self-Organizing Map

The SOM consists of a (usually two-dimensional) regular lattice or grid of map units. The most common SOM grid type is probably the hexagonal grid but a more natural choice with images is to use a rectangular grid (used also in Figures 3.1, 3.2, and 5.3). A model vector $\mathbf{m}_i \in \mathbb{R}^K$ is associated with each map unit i. The map attempts to represent all the available observations $\mathbf{x} \in \mathbb{R}^K$ with optimal accuracy by using the map units as a restricted set of models. During the training phase, the set of feature vectors is presented to the map multiple times (usually either in random sequence or in batch mode) and the model vectors stored in the map units are modified to match the distribution and topological ordering of the feature vector space. It can thus be used to visualize high-dimensional data, usually on a two-dimensional grid.

The fitting of the model vectors is usually carried out by a sequential regression process, where $t = 0, 1, 2, \ldots, t_{max} - 1$ is the step index: For each input sample $\mathbf{x}(t)$, first the index $c(\mathbf{x})$ of the *best-matching unit* (BMU) or the *winner model* $\mathbf{m}_{c(\mathbf{x})}(t)$ is identified by the condition

$$\forall i: \quad \|\mathbf{x}(t) - \mathbf{m}_{c(\mathbf{x})}(t)\| \leq \|\mathbf{x}(t) - \mathbf{m}_i(t)\| \; . \tag{3.1}$$

The usual distance metric used here is the Euclidean distance (4.4). After finding the BMU, a subset of the model vectors constituting a neighborhood centered around the BMU (node $c(\mathbf{x})$) are updated as

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + h(t; c(\mathbf{x}), i)(\mathbf{x}(t) - \mathbf{m}_i(t)) \; . \tag{3.2}$$

Here $h(t; c(\mathbf{x}), i)$ is the *neighborhood function*, a decreasing function of the distance between the ith and $c(\mathbf{x})$th nodes on the map grid. This regression is reiterated over the available samples and the value of $h(t; c(\mathbf{x}), i)$ is allowed to decrease in time to guarantee the convergence of the prototype vectors $\mathbf{m}_i$. Large values of the neighborhood function $h(t; c(\mathbf{x}), i)$ in the beginning of the training initialize the map and small values on later iterations are needed in fine-tuning.

The SOM algorithm has a number of important properties (see e.g. Haykin 1999) that make it suitable for indexing image feature data. The SOM grid provides a good approximation of the input space in a way that preserves topological ordering, which is a useful property lacking in basic clustering algorithms and especially convenient for database browsing and visualization. Image features are often characterized by high dimensionalities which are problematic for many indexing methods, especially those based on recursive partitioning of the feature space or the data points. In comparison, the SOM has a remarkable tolerance for high input dimensionalities and an innate ability to perform feature selection. In addition, the dimensionality reduction aspect of the SOM is advantageous for interactive retrieval systems due to the reduction in online computational requirements. The common property of unsupervised learning, that classes with only a small number of samples are easily lost among the predominant characteristics of the data, is present also in SOMs of limited number of map units. Still, it is often the case that this effect is less striking in the SOM when compared i.e. to linear methods like PCA or methods based on global optimization such as MDS.

After the training phase, all feature vectors are mapped to the SOM, each one to its BMU, i.e. to the map unit whose model vector is nearest to it. In image indexing, each feature vector has an associated image, so each map unit which has at least one feature vector mapped to it can then be given a visual or image label. The natural choice for this label is the image whose feature vector is nearest to the model vector of the map unit. In this manner, we can produce visualizations of the image database. Examples are shown in Figure 3.1, in which the image labels of two 16×16-sized SOMs are displayed in the SOM grid. The image database used to produce Figure 3.1 is the Columbia Object Image Library (Nene et al. 1996). The Color Layout and Edge Histogram Descriptors from the MPEG-7 standard (MPEG

2002), see Section 6.3, were used as low-level features. From the SOM grids, the topological ordering of the label images based on their color content (above) and direction of edges (below) can be clearly observed. Another example is shown in Figure 3.2, in which a SOM trained with MPEG-7 Edge Histogram descriptors of general images (from Corel Photo CDs) is displayed. This way, the SOM provides an overview of the whole database which can readily be used to aid browsing. Then, by clicking on any of the displayed images, all images associated with that map unit can be listed.

Typical applications of SOM include visualization of process states or financial results by representing the central dependencies within the data on the map (Kohonen et al. 1996). An extensive listing of SOM papers is presented in Kaski et al. (1998) and Oja et al. (2003). The SOM has been successfully applied to text documents in the *WEBSOM* document browsing and exploration tool (Honkela et al. 1997, Kohonen et al. 2000). *WEBSOM* is a means for organizing miscellaneous text documents into meaningful maps for exploration and search. It automatically organizes the documents into a two-dimensional grid so that related documents appear close to each other. Furthermore, the SOM has been applied directly to text retrieval in Lagus (2002), where the SOM is used as a filter to reduce the number of prospective documents by determining the best map units to a given query and focusing only on documents mapped to those units. An exhaustive search is then performed among the remaining documents to identify the actual best-matching documents to the query.

The first study that the author is aware of on using the SOM in image indexing was done by Zhang and Zhong (1995). They applied the SOM as a filter of unlikely relevant images based on color and texture features. Within the images mapped to the BMU, a search for k nearest neighbors is then performed. Han and Myaeng (1996) applied the SOM to image database visualization and retrieval based on a set of simple boundary shape features. Later work on using SOM in image indexing outside the PicSOM project includes Golshani and Park (1997), Ren and Means (1998), Sethi and Coman (1999), Suganthan (2002), Hussain et al. (2002), and Oh et al. (2002). The SOM has been used in the above-mentioned studies mostly to reduce the number of candidate images before a more exhaustive similarity measure is applied and for visualization purposes.

The SOM has also been used for other tasks in image retrieval. In Ma and Manjunath (1996), the SOM was used to classify and retrieve similar subimages by their textural content. In *FourEyes* (Minka and Picard 1997), the SOM was used to classify different learning problems so that each SOM unit represents a prototype learning problem with the associated image region grouping weights. The unsupervised clustering property of the SOM was used for image segmentation in Chen et al. (1999) and Ong et al. (2002). Csillaghy et al. (2000) used the SOM to classify regions of similar texture in astronomical images with an image retrieval system called *ASPECT*. In the *RETIN* system (Fournier et al. 2001b), the SOM was used to classify image pixels randomly sampled from the database images and thereby to construct a representation of the content of the database.

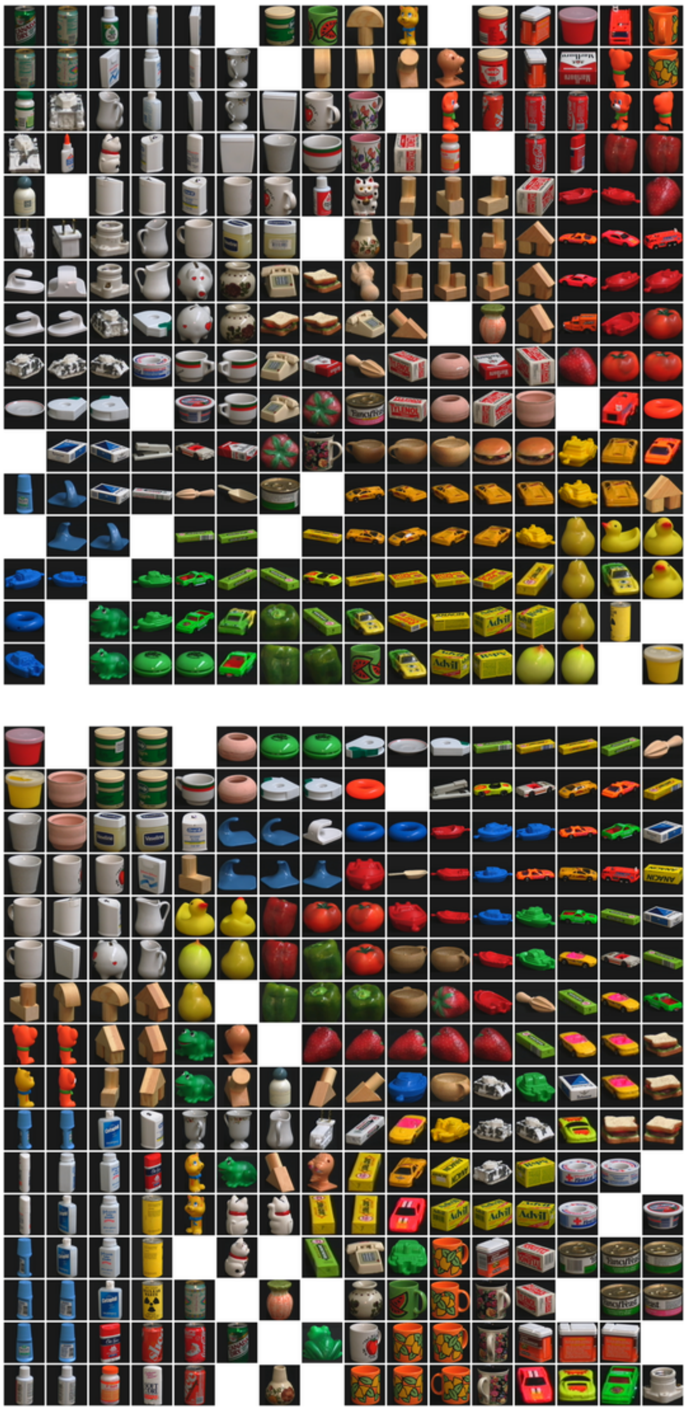

Figure 3.1: The image labels of 16×16-sized SOMs trained with Color Layout (above) and Edge Histogram (below) descriptors of the MPEG-7 standard.

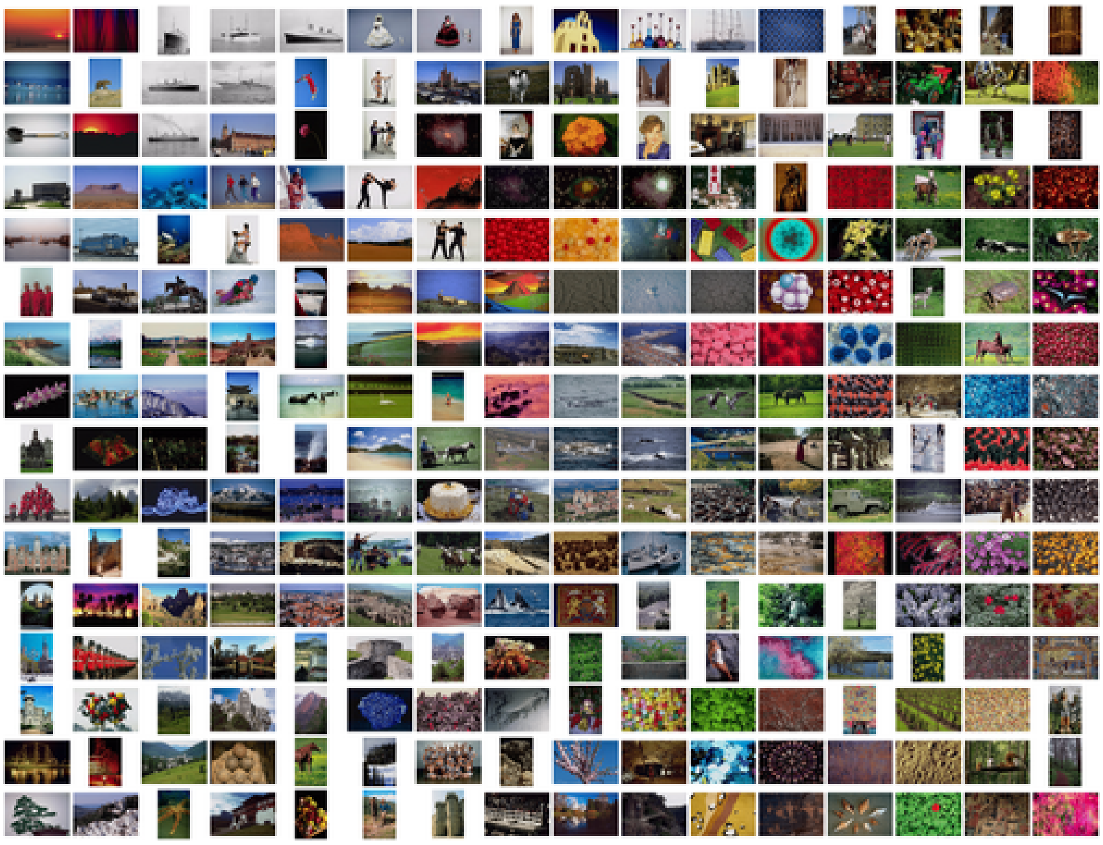

Figure 3.2: The image labels of a 16×16-sized SOM trained with the MPEG-7 Edge Histogram descriptors of general images.

3.3.2 The tree structure

The search for the BMU dominates the computing time of the SOM algorithm and it makes training large SOMs computationally too expensive especially if the dimensionality of the input vectors is high. The basic algorithm uses linear search, in which all map units must be evaluated to find the BMU. This makes the complexity of the search $\mathcal{O}(n)$, where n is the number of map units. To speed up the BMU search, Koikkalainen and Oja introduced a variant of SOM called the Tree Structured Self-Organizing Map (TS-SOM) (Koikkalainen and Oja 1990, Koikkalainen 1994). TS-SOM is a tree-structured vector quantization algorithm that uses normal SOMs at each of its hierarchical levels. It is loosely based on the traditional tree-search algorithm. Due to the tree structure, the number of map units increases when moving downwards the SOM levels of the TS-SOM. The search space for the BMU (3.1) on the underlying SOM level is restricted to a fixed-sized portion just below the BMU on the above SOM. Unlike most tree-structured algorithms, the search space does not have to be limited to the direct children of the upper level BMU. Instead, the search space can be set to include also neighboring nodes having different parent nodes in the upper level. Still, restricting the number of considered map units in the

BMU search entails the possibility of obtaining a different result than with using full search, i.e. the standard SOM algorithm. In experiments presented in Koikkalainen (1994), however, no notable differences between the results of the two algorithms were observed. The structure of a TS-SOM in one-dimensional case and the overlap of the search space in BMU search with three SOM levels is illustrated in Figure 3.3.

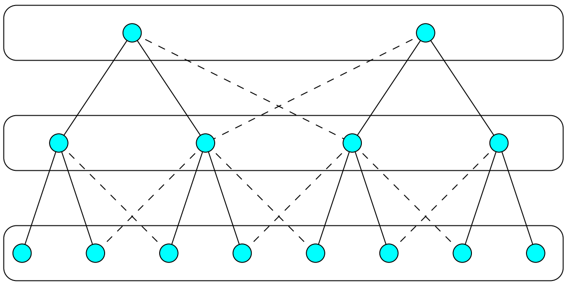

Figure 3.3: The structure of a three-level one-dimensional TS-SOM. The solid lines represent parent-child relations and the dash lines represent neighboring nodes included in the BMU search space.

The feature vectors are used to train the levels of the TS-SOM beginning from the top (smallest) level. As every TS-SOM level corresponds to a normal SOM, the training can be performed as in the standard SOM algorithm. When a level has been organized, its model vectors are frozen and the organization process advances to the next level. The upper levels are then used as a search tree to limit the search to a subset of the map units on the current level, resulting in the reduction of the time complexity of the search from $\mathcal{O}(n)$ to $\mathcal{O}(\log n)$. The complexity of the searches using TS-SOM is thus remarkably lower than if the bottommost SOM level had been accessed without the tree structure. This was confirmed also in an experiment in which TS-SOM was compared with the standard SOM (Koikkalainen 1994). In the experiment, TS-SOM was observed to be faster with networks having more than 128 map units.

The reduced computational requirements obtained by using the TS-SOM algorithm facilitate the creation and use of large SOMs, needed for indexing huge image databases. As a concrete example, calculating the TS-SOMs for the MPEG-7 descriptors (see Section 6.3 or Publication V) used in the experiments of Publications V–VII took from one and a half to 10 hours each, depending on the dimensionality of the descriptor, when an SGI Origin 2000 server equipped with 250 MHz processors was used. The used TS-SOM structure had 4 levels with sizes 4×4, 16×16, 64×64, and 256×256 map units. The data indexed by each TS-SOM consisted of 59 995 feature vectors and each vector was presented 100 times in the adaptation of each map level.

It should be highlighted that the TS-SOM differs from many approaches to producing hierarchical SOMs by the order in which the levels are trained. Hierarchical SOMs in, for example, Zhang and Zhong (1995) and Sethi and Coman (1999) are produced by starting the learning with the bottommost (largest) level and the upper levels are learned later to form a tree index to the largest SOM. Therefore, these approaches do not alleviate the computational complexity of training large SOMs but only provide a fast access to the BMU during the query phase.

4 ONLINE QUERY PROCESSING

The most important part of the functionality of a CBIR system is the processing of user requests. In this stage, the system operation is explicitly characterized by an inevitable tradeoff between system effectiveness and efficiency: the system must strive to return relevant images as accurately as possible but, on the other hand, should return its results promptly since the user is actively waiting for the retrieval algorithm to complete.

In this chapter, the main aspects of online processing in a CBIR system are described. These include measuring image similarity using either one feature or several features simultaneously, different query types, and supporting image browsing. A general system structure for reducing the computational requirements is also presented. Relevance feedback techniques are, however, bypassed in this chapter since relevance feedback forms a major topic in this thesis and will be discussed in more detail in Chapter 5.

4.1 Query specification

With low-level visual features, it is not possible to base image queries on verbal terms or other fundamental data fragments. Therefore, different query methods from those in text retrieval must generally be applied. On the other hand, a human screener can assess the relevance of an image or even a set of images to a given query very quickly with a glance whereas determining the relevance of a textual document requires much more effort.

The most common approach to formulate queries in CBIR is *query by (pictorial) example* (QBE or QBPE), the name originating from the Query-by-Pictorial-Example relational query language designed for manipulating queries with pictorial relations for retrieving LANDSAT images (Chang and Fu 1980). In QBE, the image query is based on an example or reference image shown either from the database itself (*query by internal example*) or, in some cases, the user may provide the image externally (*query by external example*). These query types have the functional difference that using an externally-provided image requires the system to index the external image on-line in order to be able to determine the similarity scores between it and the images in the database. Either way, the task of the retrieval system is then to return images as similar to the example image as possible. A closely related query type to using an external example is *query by sketch*, in which the example image is generated by the user on the fly using a sketching tool included in the retrieval interface (see e.g. Flickner et al. 1995, Del Bimbo and Pala 1997). The main problem with sketching is that users often find it difficult to produce an adequate sketch of the visual concept they are looking for. In certain restricted domains such as trademark retrieval, query-by-sketch functionality can, however, be a valuable addition. In *query by icons* or *query by visual keywords*, the example query is constructed by selecting appropriate visual elements from a prespecified selection, represented by pictorial icons, to the query. This approach is also usually infeasible in a general setting, as it requires rather sophisticated object recognition and the queries can only contain visual concepts supported by the system.

Multiple examples can be inherently supported by QBE-type queries. This is advantageous as a single example image rarely contains all and only the characterizing elements the user is looking for (Assfalg et al. 2000) whereas if the user has more than one example images to give as input, the system should be able to use them jointly and concentrate on those aspects the example images have in common. Negative examples can also be provided, highlighting undesired visual elements. In Assfalg et al. (2000), a multiple-example query is represented as a composite histogram constructed from the positive and negative examples. Zhu and Zhang (2000) presented several linear and non-linear methods for multi-example retrieval. On the other hand, the images returned by the system on earlier rounds can be considered as potential example images. This leads to relevance feedback where the user evaluates the relevance of the retrieved images and thereby guides the system toward more relevant images.

The query by example approach has also been extended for groupings of images. In *query by groups*, user-gathered groups of images are considered as the basic units of a query (Nakazato et al. 2002). In a similar manner, the query is defined by manipulating the image space by moving and grouping images by using the provided interface in the retrieval method presented by Santini and Jain (2000).

One drawback with QBE is that the success of the query considerably depends on the initial set of images as users generally do not have suitable external example images at hand. With large image databases, selecting the initially shown images is a significant problem as they should preferably contain at least one relevant image as frequently as possible. This problem is usually called the *page zero problem* (La Cascia et al. 1998). The initially shown images may be chosen so that they form an extensive coverage of the whole database, e.g. by selecting images that are as different from each other as possible or the initial images can be the result of image clustering or categorization (see e.g. Le Saux and Boujemaa 2002). Alternatively, the retrieval process may begin with a distinct browsing phase where the system shows sets of random images and the user looks for a suitable starting-point for the query phase.

A straightforward query type but only suitable to low-dimensional feature spaces in which the feature components have concrete meaning is *query by feature values*. A fixed value or a range of values is given for some or all of the feature components and images with matching feature representations are retrieved. This query type can be used e.g. for features like dominant colors and heuristic region-based shape features such as area, circularity, and elongatedness of segmented objects.

Techniques which have been used in traditional textual information retrieval would be applicable to image searching if textual descriptions of the contents of the images were available or they could be automatically produced. The latter is, unfortunately, still generally out of reach with the current state of image processing and machine vision techniques, although significant development has also been made (see e.g. Chang et al. 1998a, Naphade et al. 1998, Wang et al. 2001). As already discussed in Section 3.1.2, the image database may, however, contain such captions or other annotations either explicitly (e.g. commercial image libraries and medical databases) or implicitly (e.g. from the text surrounding an image in the WWW). If this is the case, traditional *query by keywords* can be used, either independently or in conjunction with other querying methods.

4.2 Image similarity

In a traditional database implementation, the user ordinarily makes exact queries and the items matching the query criteria are returned. Matching is a fundamental database operation, which consists of comparing the database items with the current query and deciding for each item whether or not the item satisfies the query terms. In the vector space model (VSM) framework, this query type is equivalent with a *point query*. A related query type is *range query*, in which a range of accepted values is provided for each feature. In image processing, point or range queries are used on tasks like object recognition or classification. In image retrieval, this kind of exact queries are not that useful as with general images it is difficult to find appropriate matching criteria which would pick only the relevant images. These query types are thus mostly used on retrieving images based on metadata or other non-visual features which can effectively be managed with traditional database systems.

A different approach is generally applied with visual data. Instead of matching, images are graded using a similarity criterion, resulting in a permutation of all the images in the database sorted according to the used measure of similarity. A preset number k of most-similar images are then typically presented as the query result to the user, resulting in a kNN query. Or, the result may consist of all images below a certain dissimilarity threshold α to the query. This query type is denoted as *within-distance* or *α-cut query*. An alternative basis for the image retrieval problem is to take a probabilistic approach and, instead of geometric similarity measures, consider the probabilities of images to belong to the classes of relevant and non-relevant images. This approach is discussed in Section 5.2.2.

As the goal of CBIR is to accurately retrieve images relevant to the user, the image similarity measure used in the retrieval system should correspond to the user's notion of perceptual similarity. This can only be achieved if there is enough overlap between the machine and human measures of image similarity (Squire et al. 1999b). This also emphasizes the need for certain flexibility in the similarity measure as human judgments of image similarity are subjective and context-dependent.

4.2.1 Distance measures

In addition to extracting a suitable set of features describing the contents of the images, we need a suitable measure of similarity between images in the database. For this purpose, we define a real-valued function called *global distance* $D : \mathcal{D} \times \mathcal{D} \to \mathbb{R}^+$ between all pairs of images in the database. However, since the images are in the commonly-employed VSM represented as feature vectors in a feature space, a more useful definition is the *feature-wise distance* $d : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}^+$ of two images $\mathcal{I}_i$ and $\mathcal{I}_j$ according to a K-dimensional feature $\mathbf{f}$

$$d(\mathbf{f}(\mathcal{I}_i), \mathbf{f}(\mathcal{I}_j)) = d(\mathbf{f}_i, \mathbf{f}_j) \ . \tag{4.1}$$

Feature-wise distances are typically defined using suitable metrics on the corresponding feature spaces. It should also be noted that distance is actually a measure of dissimilarity with the value of zero denoting exact match and larger values indicating less similar images whereas similarity is usually defined in the range $[0, 1]$ with 1 meaning perfect similarity and 0 no similarity. Still, distance values can easily be converted to similarity values if explicitly required and, therefore, in the following discussion we use the terms distance and similarity rather nonchalantly.

A common example of distance is the *Minkowski-form distance* based on the L_λ norm

$$d_{L_\lambda}(\mathbf{f}_i, \mathbf{f}_j) = \left[\sum_{k=1}^{K} |f_i(k) - f_j(k)|^\lambda \right]^{\frac{1}{\lambda}} \tag{4.2}$$

and thus containing a parameter λ. $f_i(k)$ is the kth component of $\mathbf{f}_i$. By setting $\lambda = 1$, we obtain the *Manhattan* or *city-block distance*

$$d_{L_1}(\mathbf{f}_i, \mathbf{f}_j) = \sum_{k=1}^{K} |f_i(k) - f_j(k)| \tag{4.3}$$

and by setting $\lambda = 2$, we get the common *Euclidean distance*

$$d_{L_2}(\mathbf{f}_i, \mathbf{f}_j) = \|\mathbf{f}_i - \mathbf{f}_j\| = \sqrt{\sum_{k=1}^{K} (f_i(k) - f_j(k))^2} \tag{4.4}$$

which is the basic metric for Self-Organizing Maps and thereby used extensively in this work.

The *generalized Euclidean distance* is defined as

$$d_{\mathrm{GE}}(\mathbf{f}_i, \mathbf{f}_j) = \sqrt{(\mathbf{f}_i - \mathbf{f}_j)^T \mathbf{A} (\mathbf{f}_i - \mathbf{f}_j)} = \sqrt{\sum_{k=1}^{K} \sum_{l=1}^{K} a_{kl} (f_i(k) - f_j(k))(f_i(l) - f_j(l))} \tag{4.5}$$

where a_{kl} is the (k, l)th element of matrix $\mathbf{A}$. Generalized Euclidean distance contains the Euclidean distance as a special case when $\mathbf{A} = \mathbf{I}$, i.e. the identity matrix.

When the values of a diagonal $\mathbf{A}$ are not equal, the isosurfaces of the generalized Euclidean distance become ellipses. If $\mathbf{A}$ is not diagonal, the generalized Euclidean distance takes correlations between feature dimensions into account. An important distance measure of this type is the *Mahalanobis distance*

$$d_{\mathrm{MA}}(\mathbf{f}_i, \mathbf{f}_j) = \sqrt{(\mathbf{f}_i - \mathbf{f}_j)^T \mathbf{\Sigma}^{-1} (\mathbf{f}_i - \mathbf{f}_j)} \tag{4.6}$$

in which $\mathbf{A} = \mathbf{\Sigma}^{-1}$ is the inverse of the covariance matrix of the feature distribution. The Mahalanobis distance is useful to limit the effect of correlations. If two feature components are strongly correlated, they capture similar characteristics of the image and this similarity is essentially taken twice into account in the above distance measures. In this sense, the Mahalanobis distance can be seen to normalize component-wise correlations in addition to normalizing variance.

Another similarity measure, widely used in text retrieval and closely related to Euclidean distance, is the *cosine measure*

$$d_{\cos}(\mathbf{f}_i, \mathbf{f}_j) = \frac{\mathbf{f}_i^T \mathbf{f}_j}{\|\mathbf{f}_i\| \, \|\mathbf{f}_j\|} = \frac{\sum_{k=1}^{K} f_i(k) f_j(k)}{\sqrt{\sum_{k=1}^{K} f_i(k)^2} \sqrt{\sum_{k=1}^{K} f_j(k)^2}} \tag{4.7}$$

which gives the same rankings as the Euclidean distance (4.4) if the vectors $\mathbf{f}_i$ and $\mathbf{f}_j$ were normalized to unit length. With normalized vectors, (4.7) can also be considered as the *correlation* between $\mathbf{f}_i$ and $\mathbf{f}_j$.

One shortcoming of the Minkowski-form distances is that the resulting distance is a function of the whole feature vectors, but the feature vectors of similar images are not always similar with respect to every feature component. Thereby, Li et al. (2002b) proposed the *dynamic partial distance function* (DPF) which takes only a subset of feature components into account when calculating feature distance. DPF is calculated as

$$d_{\mathrm{DPF}_\lambda}(\mathbf{f}_i, \mathbf{f}_j) = \left[\sum_{k \in \Delta_m} |f_i(k) - f_j(k)|^\lambda \right]^{\frac{1}{\lambda}} \tag{4.8}$$

where Δ_m is a set containing the $m \leq K$ components of the feature space with the smallest values for $|f_i(k) - f_j(k)|, \; k = 1, \ldots, K$.

Multidimensional distributions are often compressed by partitioning the multidimensional space into bins, resulting in the data represented by a histogram. Histograms are an important structure for image retrieval as many commonly-used visual features are represented as histograms. A histogram with K bins can be interpreted either as a point in a K-dimensional space or as a probability distribution. Therefore, the distance measures presented above can be applied to histograms, but distance measures suitable for histograms in particular exist. Measuring the similarity of two histograms is a significant and widely studied issue in the CBIR field (see e.g. Smith 1997, Rubner 1999, Brunelli and Mich 2001, Cha and Shihari 2002). Furthermore, a number of methods to improve the efficiency of using histograms with large databases has been presented. An exhaustive search where each image in the database is considered easily becomes a bottleneck for system scalability with respect to database size. A method for further reducing the amount of needed computations by using simple low-dimensional distance measures obtained with SVD which are lower bounds of histogram distances was presented in Hafner et al. (1995). In Berman and Shapiro (1997), the triangle inequality was utilized to eliminate unnecessary histogram comparisons. Song et al. (2001) presented a multiresolution comparison method for histograms which can remove candidate histograms whose lower bound is larger than the current minimum distance.

Histogram-based distances can be divided to two categories: *bin-by-bin* and *cross-bin* or *quadratic* distances. Bin-by-bin distances compare only the corresponding histogram bins whereas cross-bin distances compare also non-corresponding bins. In addition to the Minkowski-form distances, typical examples of bin-by-bin distances include *histogram intersection* defined by

$$d_{\mathrm{HI}}(\mathbf{f}_i, \mathbf{f}_j) = \frac{\sum_{k=1}^{K} \min(f_i(k), f_j(k))}{\sum_{k=1}^{K} f_i(k)} \ , \tag{4.9}$$

Kullback-Leibler divergence

$$d_{\mathrm{KL}}(\mathbf{f}_i, \mathbf{f}_j) = \sum_{k=1}^{K} f_i(k) \log \frac{f_i(k)}{f_j(k)} \ , \tag{4.10}$$

and *Jeffrey divergence*

$$d_{\mathrm{JD}}(\mathbf{f}_i, \mathbf{f}_j) = \sum_{k=1}^{K} \left(f_i(k) \log \frac{f_i(k)}{\hat{f}(k)} + f_j(k) \log \frac{f_j(k)}{\hat{f}(k)} \right) \tag{4.11}$$

where $\hat{f}(k) = (f_i(k) + f_j(k))/2$ is the mean histogram. Of the above distances, the first two are not true metrics as they are not symmetric. For histogram intersection, this is easily fixed by normalizing the histograms. In fact, the histogram intersection equals the Manhattan distance if the histograms are normalized (Swain and Ballard 1991). Using the Kullback-Leibler and Jeffrey divergences requires normalizing the histograms so that they sum up to unity as the measures are only meaningful to probability distributions. In contrast to the Kullback-Leibler divergence, the Jeffrey divergence is symmetric and numerically stable with empirical distributions. Nonparametric test statistics, providing a sound procedure for testing the hypothesis that two empirical distributions were generated from the same distribution, can also be used for histogram comparison (Rubner et al. 2001). While being computationally inexpensive, all bin-by-bin measures are sensitive to the selection of bin boundaries since they do not share information across the boundaries. Bin-by-bin distances are also sensitive to the dimensionality of the histogram, i.e. the selection of K.

The sensitivity problems of bin-by-bin distances can be alleviated by using cross-bin distances, which are usually defined using (4.5) with a non-diagonal $\mathbf{A}$. Then, a_{kl} is the weight associated with the similarity of bins k and l. A cross-bin histogram distance measure was suggested for color histograms in Niblack et al. (1993), in which the elements of matrix **A** were set to

$$a_{kl} = 1 - \frac{d(k,l)}{d_{\max}} \tag{4.12}$$

where $d(k,l)$ is the distance between colors k and l in the used color space and $d_{\max}$ is the maximum distance of any two colors. A basic implementation of a cross-bin distance has quadratic complexity, i.e. $\mathcal{O}(K^2)$, although it can be reduced to $\mathcal{O}(K)$ with certain precomputations.

A different approach is taken by Rubner et al. (1998) who consider histogram distance as a transportation problem. They propose the *Earth Mover's Distance (EMD)* which is based on determining the minimal cost to transform one histogram to another by moving "histogram mass". EMD is computed using a linear optimization algorithm, which makes it computationally rather heavy and thus less useful in online retrieval applications.

4.2.2 Feature selection and synthesis

Due to the gap between high-level semantics and low-level visual features, the retrieval performance of any low-level visual feature is bound to remain low at least for some retrieval tasks. On the other hand, finding a single metric which would universally capture image similarity as perceived by humans is altogether an ill-posed problem due to the inherent subjectivity. Therefore, it can be stated that effective CBIR generally requires the use of multiple features. This can also be seen as analogous to classifier combination in statistical pattern recognition. In addition to the development of improved classifiers, the performance of a pattern recognition system can be improved by using multiple classifiers in parallel and combining their responses. If the classifiers are designed to complement each other and the combination algorithm can utilize the strengths of individual classifiers, this approach can lead to superior performance. A study of using multiple classifiers in image retrieval is presented in Hsieh and Fan (2001).

Combining multiple features can be achieved with either a sequential or a parallel approach. In sequential combination or *feature filtering* each feature is invoked in a linear sequence to remove non-relevant images according to that particular feature. A natural choice is to set the order of features so that the computationally cheapest features are invoked first. After all feature filters have been applied, the final selection of images can be performed for the remaining images. However, a more common approach is to consider the features independently and in parallel, which is generally achieved by two overlapping approaches, *feature selection* and *feature weighting*, both of which can be automatic, interactive (semi-automatic), or manual. For feature selection, one solution would be to offer a wide range of object and scene intrinsic features for the user to select from. For instance, we could make available custom-made shape detectors for different objects that might be present in the images and the user would select the ones suitable for the given query. One important application for this type of approach is human face detection. In general-purpose retrieval systems with unconstrained images, however, this approach is not very practical. Generally, the discrimination abilities of different features are not evident and selecting a viable feature set for a given task is difficult. Automatic or semi-automatic methods would thus be preferable. Automatic feature selection is discussed e.g. in Breiteneder and Eidenberger (2000). Alternatively, the retrieval algorithm may be designed so that it is capable of neglecting poorly-working features and focusing on the ones providing the most useful information, in which case explicit feature selection is not needed.

Most of the early CBIR systems supporting multi-feature retrieval, such as *QBIC* and *Virage*, relied on user-provided weights for a given set of features. However, the results of the retrieval are often strongly dependent on the given values for the weights, but providing suitable ones is again a difficult task, even for a knowledgeable expert, let alone a normal user (Picard et al. 1996). Therefore, these decisions should be made automatically by the retrieval system or semi-automatically, which generally leads to interactive retrieval and learning from user interaction (discussed in Chapter 5).

In a setting supporting multiple features, the global distance D of two images is, in general, a function of feature-wise distances $d_m,\ m = 1, \ldots, M$ of the images in all M used feature spaces:

$$D(\mathcal{I}_i, \mathcal{I}_j) = g(d_1(\mathbf{f}_i^1, \mathbf{f}_j^1), d_2(\mathbf{f}_i^2, \mathbf{f}_j^2), \ldots, d_M(\mathbf{f}_i^M, \mathbf{f}_j^M)) \ . \tag{4.13}$$

Simple possibilities for the function $g(\cdot)$ are e.g. $g = \min$, $g = \max$, and $g =$ median. In many cases, however, the above definition is overly broad and the global distance is simply a linear combination of the feature distances. The above equation can thus be simplified to

$$D(\mathcal{I}_i, \mathcal{I}_j) = \sum_{m=1}^{M} W_m d_m(\mathbf{f}_i^m, \mathbf{f}_j^m) \tag{4.14}$$

where W_m is the weight parameter associated with the mth feature space. Feature combination can also be performed with other approaches such as voting and Borda count. For example, a voting procedure is presented in Nastar et al. (1998), where each feature is used to grade images separately and the final ranks of retrieved images are obtained by averaging the separate ratings. A modified Borda count method was used to combine results from multiple features in Jeong et al. (1999). In Sheikholeslami et al. (1998), a MLP network was used for this purpose.

4.3 Image browsing and database visualization

There are two general methods for finding images from large databases: querying and browsing. Until now, this section has discussed the querying approach where the retrieval consists of strictly defined alternating user-system interaction. Database browsing, on the other hand, is much more vague since it basically means some kind of free-form maneuvering by the user in the image space. The query paradigm, although being currently the method of choice in most CBIR research, has its problems and limitations. The search task itself can be indefinite, the user may just be looking for interesting images whatever they might be (open-ended search), or she may change the query target during a query. Effective query by example requires positive example images and locating one or more of them in the beginning of the query is a recognized problem (cf. Section 4.1). Images gathered during browsing can be used as initial examples for later QBE-type queries. Browsing and querying may and even should be tightly integrated so that it is possible to switch between them at any time as, for example, in Pečenović et al. (2000).

An integrated browsing tool can thus be very useful in image retrieval applications. Designing effective interfaces for browsing is, however, not at all a trivial task. Since browsing involves processing the entire database, some type of database visualization and a tool for navigation are needed. Browsing and navigating a database can be disorienting unless the user can form a mental picture of the entire database. An insight of the surrounding environment is required to be able to effectively decide where to proceed next (Rubner 1999). To assist browsing, images should be organized so that similar images are grouped together or located near each other in the visualization. Image similarity, as already discussed in Section 4.2, is a difficult issue but even organization by low-level features can be useful for browsing (Rodden et al. 2001). A common approach to provide a database overview is to use dimensionality reduction of the image features to (usually) a two-dimensional plane using techniques like PCA (e.g. Hiroike et al. 1999, Tian et al. 2002), MDS (e.g. Rodden et al. 1999, Rubner 1999, Stan and Sethi 2003) or SOM (as shown e.g. in Figures 3.1 and 3.2). The resolution of the display area is limited and therefore the original images are typically represented by small thumbnail images. Also, label images or textual labels can be used to represent groups of similar images. Distorted displays such as fish-eye lenses and magnifying glasses can also be used.

To aid browsing of large databases, a common method is to provide a hierarchical display of the database and support traversing both on the current level of the hierarchy and between levels of the hierarchy with operations such as panning and zooming. Tree Structured SOMs, as discussed in Section 3.3, have been used in PicSOM for this purpose. This is illustrated in Figure 4.1, where the image labels of a three-level TS-SOM are displayed. First, the uppermost SOM contains a cursory view of the database with only a small number of map units. The level of detail is then increased when moving downwards in the hierarchy until the full contents of individual map units can be reached from the bottommost map level. Similar approaches with hierarchical SOMs were presented in Zhang and Zhong (1995) and Sethi and Coman (1999). MDS can also be used for hierarchical browsing, for instance, by incorporating it with a clustering method as in Stan and Sethi (2003). In Chen et al. (2000), the hierarchical browsing environment is constructed using a similarity pyramid. Hierarchical clustering has also been used for this purpose (Krishnamachari and Abdel-Mottaleb 1999, Pečenović et al. 2000).

Another approach, which can be seen as a combination of querying and browsing, is

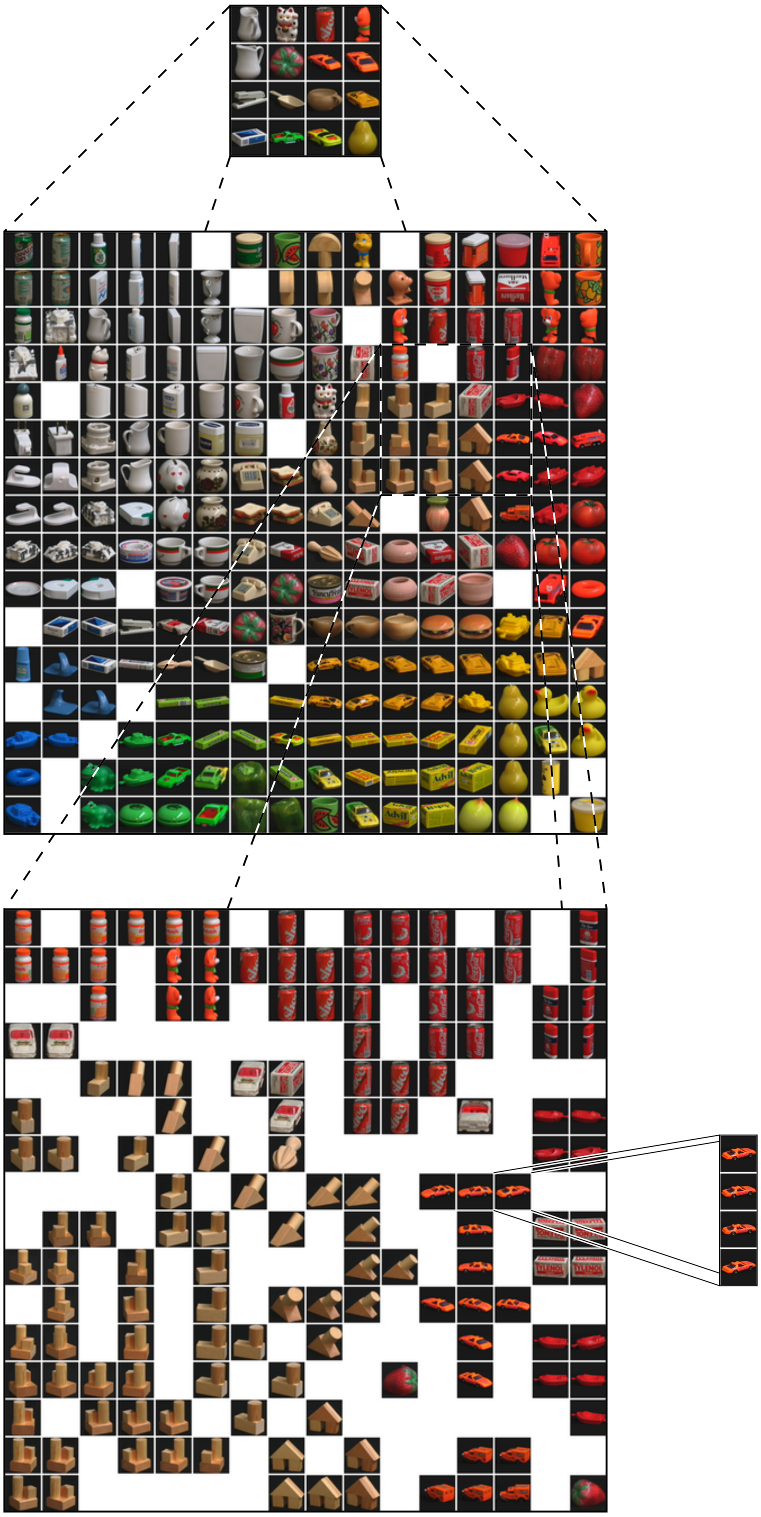

Figure 4.1: An illustration of image labels of a three-level TS-SOMs trained with MPEG-7's Color Layout descriptor. Only a small portion of the bottommost SOM is displayed due to limited space. The dashed lines indicate the hierarchy. In addition, the whole set of images mapped to one map unit on the bottommost SOM is displayed.

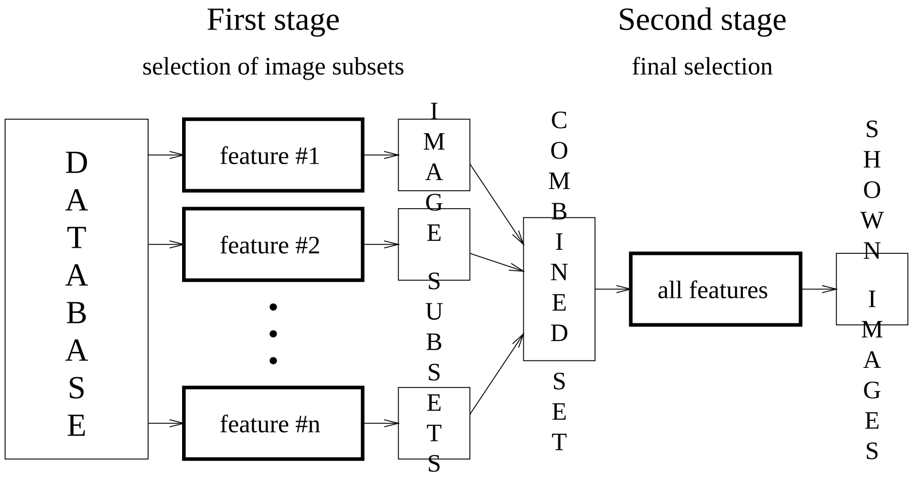

Figure 4.2: A two-stage structure for CBIR systems.

to use dynamic visualization. The user interface is designed to allow more complicated interaction and direct manipulation of images inside the visualization display, instead of just providing navigation aids in a static environment. The visualization display can be used to form groupings of relevant example images and to move non-relevant images away as in Santini and Jain (2000), Caenen et al. (2000), Nakazato et al. (2002), and Tian et al. (2002).

4.4 A generic structure of multi-feature CBIR systems

With large databases, the requirement of immediate or speedy responses during query processing often limits the amount of computations which can be performed and, in many cases, computational shortcuts must be applied. One such shortcut is to divide and conquer the image selection process by making it in multiple stages. This approach is presented and discussed in more detail in Publication IV. Figure 4.2 illustrates the idea within a two-stage structure corresponding to the parallel method for feature combination (Section 4.2.2). In a multi-feature retrieval setting, each feature representation $m = 1, \ldots, M$ can be used separately for finding a set $\mathcal{D}^{\alpha}_{m}$ of image candidates according to that feature. This is especially advantageous if the distances calculated in the different feature spaces are weighted dynamically as in such a case it is not possible to order the images by their mutual distances in advance. Considerable savings in computational requirements may be achieved if we can set $N^{\alpha}_{m} \ll N$ provided that satisfactory retrieval precision can still be reached. At minimum, the number of images in each subset, N^{α}_{m}, should exceed the count of images to be finally shown to the user.

The per-feature subsets are then combined into a larger set $\mathcal{D}^{\beta}$ of images which may be further processed in a more exhaustive manner. Depending on the sizes of the subsets, for example, the union of the initial sets, $\mathcal{D}^{\beta} = \bigcup_{m=1}^{M} \mathcal{D}^{\alpha}_{m}$, or only those which are included in more than one of them, can be taken into the combined set $\mathcal{D}^{\beta}$.

Nevertheless, the objective is that in the final selection process there will be involved a substantially smaller number of images than the whole database, i.e. $N^{\beta} \ll N$. This enables to use computationally more demanding techniques for selecting the finally shown images among the images in this set.

A variety of different CBIR techniques can be represented in terms of this kind of common system structure. In Publication IV, the PicSOM method and reference CBIR methods based on vector and scalar quantization are represented within the structure illustrated in Figure 4.2. The structure of Figure 4.2 is also compatible with the feature combination and weighting presented in (4.13) and (4.14). Generally in this setting $N_m^{\alpha} = N^{\beta} = N$, so there is no computational savings for the processing in the second stage due to a reduced number of potential images. In any event, the operation of different CBIR systems can then be analyzed by studying the functionality of these blocks.

5 RELEVANCE FEEDBACK

The iterative and interactive refinement of the original formulation of a query is known as *relevance feedback* in IR literature (Salton and McGill 1983, Baeza-Yates and Ribeiro-Neto 1999). The essence of relevance feedback is to move from one-shot or batch mode queries, as provided by standard tools like SQL, to navigational queries where one query consists of multiple rounds of interaction and the user becomes an inseparable part of the query process. During a round of relevance feedback, the user is presented with a list of retrieved items and is expected to evaluate their relevance, which information is then fed back to the retrieval system. The expected effect is that the new query round better represents the need of the user as the query is steered toward the relevant items and away from the non-relevant ones. Ever since the early experiments with the *SMART* system (Salton 1971), using relevance feedback has shown considerable improvements in retrieval precision and user interaction has remained a major topic in IR research (see e.g. Ingwersen 1992). Three distinct strengths of relevance feedback are listed by Baeza-Yates and Ribeiro-Neto (1999): (a) It shields the user from the inner details of the retrieval system. (b) It brings down the retrieval task to small steps which are easier to grasp. (c) It provides a controlled setting to emphasize some features and de-emphasize others.

In the first part of this chapter, the major variants of today's relevance feedback algorithms in CBIR are introduced. Relevance feedback has excited a formidable amount of research interest in the CBIR field, especially in the few recent years, so this chapter does not contain a comprehensive survey of all the existing and proposed techniques. Instead, the major approach types into which most of the current methods can be categorized are described. A taxonomy for the algorithms is presented, although the categorization is not a strict one; the categories of several methods are more results of the methods' point of view. Next, the chapter contains a fairly detailed description of the relevance feedback technique proposed by our research group. Finally, the chapter is concluded with a discussion on using relevance assessments recorded during normal usage of the retrieval system in a longer-term learning scheme devised by the author of this thesis.

5.1 Relevance feedback in image retrieval

Soon after the first prototype CBIR systems, relevance feedback was quickly adopted to image retrieval (see e.g. Picard et al. 1996, Rui et al. 1997, Huang et al. 1997a), where it has proven out to be widely successful and the majority of current CBIR systems include some kind of a relevance feedback mechanism. There are two main reasons for this popularity. First, more ambiguity arises in interpreting images than text, making user interaction more necessary (Zhou and Huang 2003) Second, manual modification of the initial query formulation is much more difficult in CBIR than with textual queries. Still, the research on relevance feedback in the CBIR setting can be seen as a direct descendant of general interaction research in IR. Reviews of relevance feedback techniques used in CBIR have recently been published by Zhou and Huang (2003) and Ortega-Binderberger and Mehrotha (2003).

Relevance feedback can be seen as a form of supervised learning to steer the subsequent query toward the relevant images by using the information gathered from the user's feedback. Another way to view relevance feedback in CBIR is to regard a system implementing relevance feedback as one trying to gradually learn the optimal correspondence between the high-level concepts people use and the low-level features obtained from the images. The user thus does not need to explicitly specify priorities for different similarity assessments because they are formed implicitly by the system based on the user–system interaction. This is advantageous also since the correspondence between concepts and features is temporal and case specific. This means that, in general, every image query is different from the others due to the hidden conceptions on the relevance of images and their mutual similarity and therefore using a static image similarity measure may not be sufficient. On the other hand, Santini et al. (2001) have argued that the user feedback should be seen, instead of as filtering images based on some preexisting meaning, as a process of creating meaning through the interaction. They argue that images do not have intrinsic meanings but rather the semantics of an image emerge from the context of other images and user interaction, e.g. in a CBIR setting.

In implementing relevance feedback in a CBIR system, three minimum requirements need to be fulfilled. First, the system must show the user a series of images, remember what images have already been shown, and not display them again. Thus, the system will not end up in a loop and all images will eventually be displayed. Second, the user must somehow be able to indicate which images are to some extent relevant to the present query and which are not. In this work, these images are denoted as *positive* and *negative* seen images. It is thus not sufficient that the user picks just one of the shown images, but rather a set of images must be indicated as positive ones while the remaining images can implicitly be regarded as negative ones. Clearly, this granularity of relevance assessments is only one possibility among others. In some systems, for example, the negative examples must also be explicitly provided and the non-selected images are considered to be neutral. The relevance scale may also be finer, e.g. containing options like "very relevant", "relevant", "somewhat relevant", and so on. Relevance feedback can also be in the form of direct manipulation of the query structure as with the dynamic visualization methods discussed in Section 4.3. As the third requirement, the system must change its behavior depending on the relevance scores provided for the seen images. During the retrieval process more and more images are assessed and the system has increasing amount of data to use in retrieving the succeeding image sets. The art of relevance feedback is finding the ways which use this information most efficiently.

The interactive process of relevance feedback where each seen image is classified either as positive or negative can be formalized as follows. In Section 2.1, we denoted the image database as $\mathcal{D}$ and its non-intersecting subsets of relevant and non-relevant images to a given query as $\mathcal{D}^{\oplus}$ and $\mathcal{D}^{\ominus}$, respectively. During the nth round of the retrieval session, the set of retrieved images is denoted as $\mathcal{D}_n$ and the cumulative set of retrieved images since the beginning of the query as $\mathcal{D}(n)$. After user interaction on round n, the images in $\mathcal{D}_n$ will have attached relevance assessments, i.e. the images are split into two sets, $\mathcal{D}_n^+$ and $\mathcal{D}_n^-$. The interpretation of these sets is to some extent ambiguous. A common assumption, especially in automated performance evaluation, is *categorical feedback*, where the images are assumed to be rated either as relevant or non-relevant according to whether the image in question belongs to the same image category as the target of the query. In this case, $\mathcal{D}_n^+ \subset \mathcal{D}^{\oplus}$ and $\mathcal{D}_n^- \subset \mathcal{D}^{\ominus}$. Alternatively, we can consider the set $\mathcal{D}_n^+$ as seen images that are, at this time instance, more similar to the current target than the others but not necessarily relevant in the final assessment (Cox et al. 2000). The sets $\mathcal{D}_n^+$ and $\mathcal{D}_n^-$, gathered since the beginning of the query, are the basis for query improvement by relevance feedback. Often, the sets containing images with similar relevance assessments are combined, i.e. information of the round in which a certain image was shown is neglected. Assuming categorical feedback, this results in non-intersecting sets of all positive and all negative seen images, denoted as $\mathcal{D}^+(n) = \bigcup_{i=1}^{n} \mathcal{D}_i^+ \subset \mathcal{D}^{\oplus}$ and $\mathcal{D}^-(n) = \bigcup_{i=1}^{n} \mathcal{D}_i^- \subset \mathcal{D}^{\ominus}$, respectively. The still unseen images can then be marked as $\mathcal{D}'(n)$, which leads to $\mathcal{D}'(n) = \mathcal{D} \setminus \mathcal{D}(n) = \mathcal{D} \setminus (\mathcal{D}^+(n) \cup \mathcal{D}^-(n))$. The symbol N with the same superscripts and subscripts is used for denoting the cardinalities of the respective image sets.

Three specific characteristics of relevance feedback, distinguishing it from many other applications of machine learning, were identified by Zhou and Huang (2003): (a) Small number of training samples. Compared to many supervised learning tasks, the number of samples relative to the dimensionality of the feature spaces is very small in relevance feedback. Only a rather small number of images (typically $N_n < 30$) is usually evaluated on one round of the query and users are often impatient and unwilling to provide much feedback. This makes many traditional inductive learning methods ill-suited since they fail to produce stable results. (b) Asymmetry of the training data. All images in $\mathcal{D}^+(n)$ are relevant in some specific way but every image in $\mathcal{D}^-(n)$ is non-relevant in its own way. Therefore, while $\mathcal{D}^+(n)$ may be a reasonable sample of relevant images, $\mathcal{D}^-(n)$ usually cannot represent the distribution of all non-relevant images well. (c) Real-time processing requirements. Relevance feedback is used when the user is interacting with the system and thus waiting for the completion of the algorithm. An image query may well take several rounds until the results are satisfactory, so fast response time is essential. With large databases, this usually limits the range of possible methods to ones which do not rely on processing the whole database on each query round.

5.2 Methods adopted from text-based information retrieval

Text-based IR has been intensively studied for more than forty years and the usefulness of relevance feedback has been long recognized in the research field. Therefore, a natural basis for developing relevance feedback techniques for CBIR is to study the methodology of IR and apply suitable methods in image retrieval. Two such main approaches exist, one based on the vector space model (VSM) and the other on building a probabilistic model for text retrieval. Use of these approaches in image retrieval is briefly discussed next.

5.2.1 Vector space model based methods

In VSM, each database item is represented as a point in K-dimensional space. Textual documents are commonly represented by the words they contain using the bag of words model (see Section 3.2.5). This information is then encoded into a term-by-document matrix $\mathbf{X}$. Similarly to the database items, the query is also represented as a point or vector $\mathbf{q}$ in the same vector space. In order to do retrieval, the documents are ranked according to their similarity to $\mathbf{q}$. In text-based retrieval, the standard similarity measure here is the cosine measure (4.7).

The dimensionality of the data in VSM equals the number of distinct terms present in the corpus after preprocessing. Dimensions of the data are typically reduced in the preprocessing step by removing the most common terms from the data. Overly rare terms can also be removed. Still, the remaining dimensions of the data may still well be in the order of thousands: $\mathcal{O}(10^4)$ dimensions are typical for large corpora. Fortunately, $\mathbf{X}$ is typically very sparse and it is thereby feasible to utilize inverted files (Section 3.2.5). An alternate method to reduce on-line query evaluation time is to perform dimensionality reduction on the term-by-document matrix $\mathbf{X}$. Latent semantic indexing (LSI) (Deerwester et al. 1990), i.e. applying singular value decomposition on $\mathbf{X}$, is a very common method to perform dimensionality reduction in IR.

In the VSM framework, two general methods for query improvement or reformulation exist, namely *query point movement* and *feature (component) re-weighting.* These will be briefly discussed next.

Query point movement. Since the query is represented as a query point $\mathbf{q}$ in the vector space, a straightforward approach is to relocate $\mathbf{q}$ based on the new information obtained with relevance feedback about the relevancy of nearby data items. The basic idea is to move the query point toward the part of vector space where the relevant documents are located. In image retrieval with QBE queries this can be seen as transforming the original example image to a virtual query image with statistics matching to the location of the new query point.

A classical method for moving the query point based on positive and negative examples is the Rocchio's formula (Rocchio 1971):

$$\mathbf{q}_{n+1} = \alpha \mathbf{q}_n + \frac{\beta}{N^+(n)} \sum_{\mathcal{I}_j \in \mathcal{D}^+(n)} \mathbf{f}_j - \frac{\gamma}{N^-(n)} \sum_{\mathcal{I}_j \in \mathcal{D}^-(n)} \mathbf{f}_j \tag{5.1}$$

where $\mathbf{q}_n$ is the query point on the nth round of the query and α, β, and γ are weight parameters ($\alpha+\beta+\gamma = 1$) controlling the relative importance of the previous query point, the average of relevant images, and the average of non-relevant images, respectively. Usually, the information contained in the relevant images is more valuable than the information provided by the non-relevant ones. This is due to the fact that the relevant items can be reasonably assumed to be concentrated on a specific area of the vector space whereas the non-relevant items are often more heterogeneous. Therefore, we should set the weights so that $\beta > \gamma$. Setting $\gamma = 0$ is also possible, resulting in purely positive feedback. It is also plausible to set $\alpha = 0$ which results in ignoring the query history including the user-provided initial query and setting the new query point solely based on the currently available relevance assessments. In practice, the original query often contains important information which should not be neglected (Salton and McGill 1983).

Early implementations of relevance feedback via query point movement in CBIR include Rui et al. (1997), Huang et al. (1997a) and Chua et al. (1998). The effect of positive and negative feedback for query point movement in image retrieval was studied by Müller et al. (2000a). In their experiments using negative feedback improved the results, although care must be taken not to incorporate too much negative feedback to the query.

Feature component re-weighting. The basic idea is to increase the importance of those components (dimensions) of the used feature vectors which seem to aid the most in retrieving relevant images. Each component in a feature representation can be given a weight which is used in calculating the distances between images. This can be easily done by augmenting the used distance measure (see Section 4.2.1) with component-wise weights: the weight of kth component of the mth feature is denoted as w_{mk}. These weights should not be mixed with feature weights (the weight of mth feature was denoted as W_m in Section 4.2.2) but rather seen as an extension of that approach to a lower level in the similarity measurement hierarchy. As an example, the general form of Minkowski-type distances becomes

$$d_{wL_\lambda}(\mathbf{f}_i, \mathbf{f}_j) = \left[\sum_{k=1}^{K} w_{mk} |f_i(k) - f_j(k)|^\lambda \right]^{\frac{1}{\lambda}} \tag{5.2}$$

when augmented with component-wise weights (compare with (4.2)).

Assuming that the feature components are independent, the case when the relevant items have similar values for $f(k)$, i.e. the kth component of feature $\mathbf{f}$, it can be assumed that $f(k)$ captures something the relevant items have in common and which corresponds to the user's information need. Conversely, a component which has a wide spread of values for the relevant items is likely to be a poor descriptor. Therefore, an intuitive weighting scheme is to use the inverse of the standard deviation of the relevant items as component-wise weights

$$w_{mk} = \frac{c}{\sigma_{mk}^{+}} \ . \tag{5.3}$$

where c is a normalizing constant, often set so that $\sum_{k=1}^{K} w_{mk} = 1$. This scheme is used by many CBIR researchers (e.g. Rui et al. 1997, Yang et al. 1998, Aksoy et al. 2000, Brunelli and Mich 2000, Wu and Manjunath 2001). Naturally, other decreasing functions of σ_{mk}^{+} could also be applied. The estimation of standard deviation requires at least two relevant items and the estimate may well be inaccurate until enough training samples have been obtained, which is a well-known problem in relevance feedback. Before using (5.3), the feature should be normalized so that equal emphasis is placed on components with different ranges of values. A common solution is to use Gaussian normalization

$$f(k)^{\text{norm}} = \frac{f(k) - \mu_k}{K\sigma_k} \tag{5.4}$$

where K is a parameter controlling the probability that a feature component value lies inside a specific range after normalization. E.g., assuming a Gaussian distribution, 68% of the samples lie in $[-1, 1]$ with $K = 1$. For an extensive discussion on feature normalization in CBIR, see e.g. Aksoy and Haralick (2001).

The inverse standard deviation weighting of (5.3) neglects the negative seen images and several extensions for including them have been proposed. For example, the weighting of feature components can be made dependent on the difference of the inverse variances of the positive and all shown images (Schettini et al. 1999) or the component's ability to separate the positive and negative examples can be measured using the difference of their means (Yang et al. 1998) or the distribution pattern of the negative examples (Wu and Zhang 2002). Doulamis and Doulamis (2001) presented a weighting scheme where the importance of each component is estimated by simultaneously maximizing the correlation (4.7) between the query point and the positive examples and minimizing the correlation with the negative examples.

Upper-level weights can be updated using a similar approach. In Section 4.2.2, feature weights W_m (4.14) were discussed as a method to combine several features into an image query. Relevance feedback provides a way to automatically infer the W_ms based on how well the corresponding feature seems to work in the current query. A three-level weighting model was presented in Rui et al. (1998): separate weights are used at feature, representation, and component levels. In their terminology, a "feature" corresponds to modalities of low-level features (e.g. color) whereas "representations" are specific means to compute that feature (e.g. color histogram, dominant colors). All weights are updated via relevance feedback. A similar multi-level weighting approach and a back-propagation algorithm for weight updating was presented by Fournier et al. (2001a).

A method incorporating both query point movement and feature component reweighting was proposed by Ishikawa et al. (1998). The method also supports incorporating correlations between feature components. The starting point in their method is the generalized Euclidean distance (4.5). The user's relevance assessments are used to estimate both the coefficients of the matrix $\mathbf{A}$ (i.e. the implied distance function) and the optimal query point $\mathbf{q}$. This is done by solving the optimization problem

$$\begin{aligned} \min_{\mathbf{q},\,\mathbf{A}} \quad & \sum_{\mathcal{I}_j \in \mathcal{D}(n)} v_j (\mathbf{f}_j - \mathbf{q})^T \mathbf{A} (\mathbf{f}_j - \mathbf{q}) \\ \text{s.t.} \quad & \det(\mathbf{A}) = 1 \end{aligned} \tag{5.5}$$

where $[v_1, \ldots, v_{N(n)}]^T$ is a vector containing the user's relevance scores for the seen images. They also showed that inverse standard deviation weighting (5.3) gives the optimal solution to (5.5) if $\mathbf{A}$ is restricted to a diagonal matrix. Despite its theoretical appeal, the method is not feasible in practice since it requires much more data than is available in typical settings where relevance feedback is applied. When estimating both $\mathbf{A}$ and $\mathbf{q}$, the number of parameters is inevitably high compared to the number of typically available relevance assessments in relevance feedback, especially if $\mathbf{A}$ is not restricted to a diagonal matrix.

5.2.2 Probabilistic model

The probabilistic model is another classical model in information retrieval (Salton and McGill 1983, Baeza-Yates and Ribeiro-Neto 1999). Now, the retrieval problem is expressed within a framework provided by probability theory. Each database item $\mathcal{I}$ is associated with the estimated probabilities $P(\mathcal{I} \in \mathcal{D}^{\oplus})$ and $P(\mathcal{I} \in \mathcal{D}^{\ominus})$. Items with the highest probabilities to belong to the ideal answer set $\mathcal{D}^{\oplus}$ are then returned as the query result.

Relevance feedback can be incorporated to this framework by introducing the query history $\mathcal{H}_n$ where n is the number of the query round. $\mathcal{H}_n$ consists of the images displayed on query rounds up to round n $(\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n)$ and the corresponding actions $\mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n$ taken by the user. Often, the action $\mathcal{A}_j$ consists only of marking the relevant items of the returned ones, in which case we can write $\mathcal{A}_j = \mathcal{D}_j^{+}$. Generally, $\mathcal{H}_n = \{\mathcal{D}_1, \mathcal{A}_1, \mathcal{D}_2, \mathcal{A}_2, \ldots, \mathcal{D}_{n-1}, \mathcal{A}_{n-1}, \mathcal{D}_n, A_n\}$ and the probabilities of database items being relevant or non-relevant to the query given the session history are written $P(\mathcal{I} \in \mathcal{D}^{\oplus} \,|\, \mathcal{H}_n)$ and $P(\mathcal{I} \in \mathcal{D}^{\ominus} \,|\, \mathcal{H}_n)$. The task is then to determine $\mathcal{D}_{n+1}$.

In the CBIR field, the probabilistic model was first used by Cox et al. in their *PicHunter* CBIR system (Cox et al. 1996, Cox et al. 2000). In their basic formulation, they considered only target search (cf. Section 2.1) so $\mathcal{D}^{\oplus} = \{\mathcal{I}^{\oplus}\}$. Therefore, we only need to compute $P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_n)$ for all unseen images in the database. Using Bayes' rule we can write

$$P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_n) = \frac{P(\mathcal{H}_n \,|\, \mathcal{I} = \mathcal{I}^{\oplus}) P(\mathcal{I} = \mathcal{I}^{\oplus})}{P(\mathcal{H}_n)} \tag{5.6}$$

where $P(\mathcal{H}_n \,|\, \mathcal{I} = \mathcal{I}^{\oplus})$ is the likelihood of the history given that $\mathcal{I}$ is the target image and $P(\mathcal{I} = \mathcal{I}^{\oplus})$ is the *a priori* probability of $\mathcal{I}$ being the target image. In

PicHunter, $P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_n)$ is computed incrementally from $P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_{n-1})$ with

$$P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_n) = P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{D}_n, \mathcal{A}_n, \mathcal{H}_{n-1}) = \frac{P(\mathcal{A}_n \,|\, \mathcal{I} = \mathcal{I}^{\oplus}, \mathcal{H}_{n-1}) P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_{n-1})}{\sum_{\mathcal{I} \in \mathcal{D}} P(\mathcal{A}_n \,|\, \mathcal{I} = \mathcal{I}^{\oplus}, \mathcal{H}_{n-1}) P(\mathcal{I} = \mathcal{I}^{\oplus} \,|\, \mathcal{H}_{n-1})} \tag{5.7}$$

where the likelihood functions are written without the variable $\mathcal{D}_n$ as it is a deterministic function of $\mathcal{H}_{n-1}$. The term $P(\mathcal{A}_n \,|\, \mathcal{I} = \mathcal{I}^{\oplus}, \mathcal{H}_{n-1})$, referred to as the *user model*, is a critical component of the method as it models the user's response given the query history $\mathcal{H}_{n-1}$ and the target image $\mathcal{I}^{\oplus}$.

Bayesian relevance feedback for category search has been studied by Vasconcelos and Lippman (1999). They presented a similar technique as above to determine the posterior probabilities for images to belong to the class of relevant images to minimize the probability of retrieval error. Later work based on the *PicHunter* framework include Geman and Moquet (1999), Müller et al. (1999), and Su et al. (2001).

5.3 Other relevance feedback techniques in CBIR

5.3.1 Set-theoretic machine learning

Picard et al. were among the first to study learning from user interaction in image retrieval (Picard et al. 1996, Minka and Picard 1997). Their *FourEyes* system first forms initial within-image and across-image groupings of related image regions by hierarchical clustering. User feedback is then used for set-theoretic machine learning of new rules with three classical algorithms: set covering, decision list, and decision tree. In set covering, the task is to find a set of groupings which cover as much positive examples as possible but not any negative examples. In decision list and decision tree algorithms, the asymmetry of set covering is removed by allowing also sets of only negative examples and the use of set complements.

5.3.2 Density estimation

The purpose of relevance feedback can be viewed as a task of probability density estimation. In fact, the VSM-based methods described in Section 5.2.1 can be considered as density estimation of relevant images with the assumption of a unimodal Gaussian distribution (Wu et al. 2002). Given a set of positive example images $\mathcal{D}^+(n)$ provided by the user and using parametric density estimation, the task is to estimate the probability density of relevant images $p(\mathbf{x} \,|\, \mathcal{D}^+(n)\,;\, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ contains the parameters of the distribution. In order to make the estimation feasible during query-time, simplifying assumptions have to be made. In Nastar et al. (1998), the feature components are assumed to be independent and their distribution is assumed to be unimodal and Gaussian. The images to be shown on the next round are determined using modified maximum likelihood estimation which takes into account also the set of non-relevant seen images $\mathcal{D}^-(n)$. In Meilhac and Nastar (1999), the assumption that the distribution of relevant images is single Gaussian is lifted and replaced by non-parametric and multimodal density estimation using Parzen windows. In this approach, a spherical kernel function $\mathcal{K}(\cdot)$ is centered on each of the positive examples and the density of relevant images is estimated as

$$p(\mathbf{x} \,|\, \mathcal{D}^+(n)) = \frac{1}{N^+(n)} \sum_{\mathcal{I}_j \in \mathcal{D}^+(n)} \mathcal{K}(\mathbf{x} - \mathbf{f}_j) \ . \tag{5.8}$$

Another approach to estimate the probability density of relevant images without the assumption of unimodality is to use mixture models, of which the most popular is the Gaussian mixture model (GMM)

$$p(\mathbf{x} \,|\, \mathcal{D}^+(n) \,;\, \boldsymbol{\theta}) = \sum_{i=1}^{G} \pi_i \mathcal{N}(\mathbf{x} \,|\, \mu_i, \boldsymbol{\Sigma}_i) \tag{5.9}$$

where G is the number of Gaussian mixtures $\mathcal{N}(\cdot)$ and π_i the mixing parameter satisfying $\sum_{i=1}^{G} \pi_i = 1$. Estimation of the GMM parameters $\boldsymbol{\theta} = \{\pi_i, \mu_i, \boldsymbol{\Sigma}_i\}$, $i = \{1, 2, \ldots, G\}$ is a nontrivial task for which the standard method is to use the Expectation-Maximization (EM) algorithm (Dempster et al. 1977). GMM-based relevance feedback techniques using the EM algorithm have been proposed by Vasconcelos and Lippman (1998), Yoon and Jayant (2001), and Najjar et al. (2003). In Qian et al. (2002), the EM algorithm was replaced by a method based on hypersphere coverings of the relevant images in the feature space. Unlike the ones based on the EM algorithm, their method is also able to estimate G, the number of mixtures.

The probability density of relevant images can also be estimated using Support Vector Machines (SVMs) (Vapnik 1995). Chen et al. (2001) presented a relevance feedback scheme based on a one-class SVM (1-SVM) which fits a tight hypersphere in a nonlinearly transformed feature space to include as much positive examples as possible. Due to the nonlinear transform, implemented using a Gaussian kernel function, the model is able to capture also multimodal probability densities.

5.3.3 Classification methods

The methods of feature component re-weighting (Section 5.2.1) and density estimation, described above, consider only positive example images in the basic form, although extensions which include negative examples have also been introduced. An alternate approach to take negative images into account is to treat relevance feedback as a problem of classification. The problem of image retrieval is now considered as a standard two-class classification problem: to separate the class of positive examples from the class of negative examples. As these classes are not usually linearly separable, some kind of a nonlinear classification method must be applied. However, the actual goal here is different from classification; instead of predicting class labels of input vectors, we are interested in finding more images belonging to the class of positive images.

Asymmetry of the training data was identified as one of the distinct characteristics of relevance feedback in Section 5.1. This is manifested here in the compactness of the classes, as the positive images are typically concentrated to certain area or areas in the feature space and the negative images are scattered. In Huang and Zhou (2001), this is taken into account by defining a biased classification problem in which an unknown number of classes is assumed but the user is only interested in one of them. They formulate relevance feedback as a classification problem where the classes are separated by kernel-based nonlinear discriminant functions. SVMs (Hong et al. 2000, Zhang et al. 2001) and a discriminant EM algorithm (Wu et al. 2000b) have also been proposed for this purpose. Other methods for classification-based relevance feedback include the use of boosting (Tieu and Viola 2000, Guo et al. 2002), decision trees (MacArthur et al. 2000), and RBF networks (Qian et al. 2003)

5.3.4 Selecting the distance metric

One method to apply relevance feedback is to use it to select the used distance metric (Taycher et al. 1997). Since it is difficult to determine which Minkowski distance measure $d_{L_\lambda}(\mathbf{f}_i, \mathbf{f}_j)$ (4.2) is best for a particular query, the value of λ is selected so that the smallest attainable relative distance between the positive seen images is obtained. Their *ImageRover* system introduced a relevance feedback algorithm which, in addition to estimating subvector-wise weights, also selects the distance metric to be used.

5.3.5 Active learning

Active learning refers to methods in which the learning machine actively selects the unlabeled samples to query the teacher for their labels in order to maximize information gain. Active learning has also been applied to relevance feedback (Tong and Chang 2001, Li et al. 2001), where it has a fundamental difference with the other relevance feedback techniques presented in this chapter as here the aim is not to retrieve the most probably relevant images but rather the most informative ones. Selecting the most informative images has also been studied by Cox et al. (2000), who present a method where the shown images are selected in order to minimize the expected number of future iterations in the query. In King and Jin (2001), the maximum entropy principle was used to determine the most informative images.

5.4 Relevance feedback with SOMs

Most of the relevance feedback techniques described in this chapter treat the feature space more in a global than local manner. This global attitude is manifested, for example, in linear weighting of the distances along individual feature components discussed in Section 5.2.1. However, a distance measure or feature weighting which is advantageous in the vicinity of a set of images similar to each other, may not produce favorable results for the rest of the images. Rules which are applicable in one part of the feature space are not as such generalizable to handle the whole space due to the inherent nonlinear nature of image similarity (Santini and Jain 1999).

The assumption of a single query point and decreasing relevance of images as the distance to the query point increases does not generally capture high-level semantics well due to the semantic gap, no matter how complex or efficient distance function is used. This problem is common for all methods assuming a unimodal probability density for the relevant images. A number of solutions to this problem have already been introduced in the previous sections, including the use of nonparametric density estimation with Parzen windows or techniques like GMM or SVM to model multimodal probability densities.

We now turn our attention into describing how relevance feedback can be implemented by using multiple Self-Organizing Maps. The introduced technique is the backbone of our PicSOM CBIR system and has been tested with numerous feature extraction methods and various databases; in the publications included in this thesis and also e.g. in Laakso et al. (2001), Brandt et al. (2002), Iivarinen and Pakkanen (2002), and Sjöberg et al. (2003). Contrary to most of the existing methods, the presented relevance feedback technique is local in the sense that it operates only in the local neighborhoods of images marked positive or negative by the user. Therefore, the method respects better the nonlinear nature of image similarity. On the other hand, our method dynamically produces an implicit weighting of the different features so that those features which seem to perform better than the others in that particular task are given more weight. The feature combination ability of the method was studied in Publication I. In the experiments of the article, it was seen that the proposed method is able to effectively utilize a set of parallel SOMs so that the combined retrieval result exceeds the performance of any of the features used separately.

5.4.1 Generating the sparse value fields

The PicSOM system presents the user a set of images she has not seen before on each round of the image query. The user is then expected to mark the relevant images as positive, and the system implicitly interprets the unmarked images as negative. As all images in the database have been previously mapped in their best-matching SOM units (BMUs) at the time the SOMs were trained (see Section 3.3.1), it is now easy to locate the positive and negative images on each SOM (or each level of every TS-SOM) in use. The map units are awarded a positive score for every positive image mapped in them resulting in an attached positive impulse. Likewise, associated negative images result in negative scores and impulses. These positive and negative scores are scaled so that the total sum of all scores on each map is equal to zero. If the total numbers of positive and negative images are $N^+(n)$ and

$N^-(n)$ at query round n, the positive and negative scores are

$$x_+(n) = \frac{1}{N^+(n)} \quad \text{and} \quad x_-(n) = -\frac{1}{N^-(n)} \ . \tag{5.10}$$

This way, we obtain a zero-sum sparse value field on every SOM in use.

The system remembers all responses the user has given since the query was started in these sparse value fields. The cumulative query history and the user's opinions on the relevance of shown images thus become stored in every SOM in the system. If a particular SOM unit has been the BMU for many positive images and for none or only few negative ones, it can be deduced that its content coincides well with the user's opinion. By assumption the neighboring SOM units are similar to it and the images mapped in them can likewise be supposed to be relevant for the user.

Each TS-SOM has been trained with a different feature extraction method and therefore the resulting sparse value fields are different on different SOMs. Some feature extractions may spread the responses evenly all over the map surface, resulting in a seemingly random distribution of impulses on the map. Other features may, however, cluster the positive responses densely in certain area or areas of the map. The latter situation can be interpreted as being an indication on the good performance of those particular features in the current query. The denser the positive responses are the better the feature coincides in that specific area of the feature space with the user's perception on image relevance.

Now, these three factors, namely (a) the degree of the separation of the positive and negative images on the SOM, (b) the relative denseness of the positive images, and (c) the similarity of images in neighboring map units, can be accounted for in a single action. This joint action is low-pass filtering of the sparse value fields on the two-dimensional map surfaces. This way, strong positive values from dense relevant responses get expanded into neighboring SOM units, whereas weak positive and negative values in the map areas where the responses are sparse and mixed cancel each other out. What follows in the low-pass filtering is the polarization of the entire map surface in areas of positive and negative cumulative relevance.

5.4.2 Shift-invariant window functions

Spreading of the response values can be performed by convolving the sparse value fields with a tapered (or rectangular) window or kernel function. The one-dimensional convolution of a discrete-time signal $x[n]$ and window $w[n]$ of length $L = 2l+1$ is a basic signal processing operation defined as

$$y[n] = x[n] * w[n] = \sum_{k=-l}^{l} x[n-k]w[k] \ . \tag{5.11}$$

On SOM surfaces the convolutions have to be two-dimensional. Due to computational reasons this is best implemented as one-dimensional horizontal convolution followed by one-dimensional vertical convolution (or vice versa). This can be done assuming the used convolution kernel is separable and shift-invariant. The following one-dimensional window functions were experimented with in Publication VI ($n = -l, -l+1, \ldots, l$):

$$w_r[n] = 1 \qquad \text{(rectangular)} \qquad (5.12)$$

$$w_t[n] = \frac{l - |n|}{l} \qquad \text{(triangular)} \qquad (5.13)$$

$$w_g[n] = e^{-(\frac{n}{\alpha})^2} \qquad \text{(truncated Gaussian)} \qquad (5.14)$$

$$w_x[n] = e^{-\frac{|n|}{\beta}} \qquad \text{(truncated exponential)} \qquad (5.15)$$

The truncated Gaussian and exponential windows above require a parameter (denoted here as α and β, respectively) controlling the decay of the window. For example, the parameters α and β were selected so that $w_g[\pm\frac{l}{2}] = w_x[\pm\frac{l}{4}] = \frac{1}{2}$ in Publication VI.

The length of the window is a predominant parameter of the method regardless of the used window function. With small l, the search expands only to the immediate neighbors of the relevant items and the search area widens as l grows. As the computational complexity of the convolution is linearly dependent on the window length, it is beneficial to be able to use as small windows as possible.

5.4.3 Location-dependent window functions

Information on the distances between neighboring SOM model vectors in the feature space has earlier been used mainly in visualization. The average relative distance of a model vector to its neighbors can be color-coded with gray-level shades or pseudocolors, resulting in a SOM visualization known as the *U-matrix* (Ultsch and Siemon 1990, Kraaijveld et al. 1992). Dark or dim shades are often used to visualize long distances whereas bright colors correspond to close similarity between neighboring model vectors. Especially the clustering ability of the SOM can be illustrated in this manner (see e.g. Vesanto and Alhoniemi 2000).

The relative distances between neighboring SOM units can also be utilized in our setting, leading to an alternative method for spreading the relevance responses on the SOMs. In this method, we relax the property of symmetry in the window function. Intuitively, if the relative distance of two SOM units in the original feature space is small, they can be regarded as belonging to the same cluster and, therefore, the relevance response should easily spread between the neighboring map units. Cluster borders, on the other hand, are characterized by large distances between map units and the spreading of responses should be less intensive.

For each neighboring pair of map units according to 4-neighborhood, say i and j, the distance in the original feature space is calculated. The distances are then scaled so that the average neighbor distance is equal to one. The normalized distances d_{ij} are then used for calculating location-dependent convolutions with two alternative

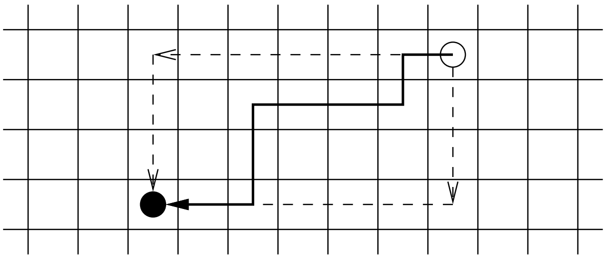

Figure 5.1: An illustration of the two methods for calculating the location-dependent convolutions on the SOM grid. In the path method (solid line), the minimum path $\circ \rightarrow \bullet$ is solved with dynamic programming. In the sum method (dashed lines), horizontal and vertical one-dimensional location-dependent convolutions are calculated in both orders and then averaged.

methods, illustrated in Figure 5.1. The *path method* uses dynamic programming to solve the minimum path length along the 4-neighborhood grid between two arbitrary map units. Given a maximum allowed distance l, we can calculate and tabularize the between-node distances d_{ij} for non-neighboring map units i and j. Then the two-dimensional convolution functions can be formed from (5.12)–(5.15) by setting $n = d_{ij}$. In the alternative *sum method*, a computationally faster solution is obtained by performing one-dimensional location-dependent convolution first horizontally with kernel values obtained again from (5.12)–(5.15) with $n = d_{ij}$. The result of the horizontal convolution is then similarly convolved with vertical one-dimensional location-dependent kernels. As the order of the successive one-dimensional convolutions now matters, the original impulse-valued SOM surface is convolved again, now first vertically and then horizontally, and the two convolution results are averaged.

Figure 5.2 illustrates how the positive and negative impulses on a sparse value field, displayed with red and blue map units on a neutral (white) background, are first mapped on a 16×16-sized SOM and how the responses are expanded into "relevance landscapes". Shift-invariant convolution is obtained with a fixed window function, such as the ones presented in (5.12)–(5.15). In Figure 5.2, triangular window (5.13) with $l = 4$ was used. Location-dependent convolution includes information about the relative distances between neighboring SOM units.

5.4.4 Combining multiple features

If we limit our consideration to only one feature or SOM, the unseen images mapped to SOM units which have the strongest positive score after the low-pass filtering are the obvious candidate images to be shown to the user on the next round. This can be easily extended to multiple SOMs. As the response values of map units of different SOMs are mutually comparable, we can determine a global ordering to find the overall best candidate images. By locating the corresponding images in all the SOMs, we get their scores with respect to different feature extraction methods. To perform a comprehensive evaluation, the scores of all images on every map should be determined. For computational reasons, this is not usually performed. Instead,

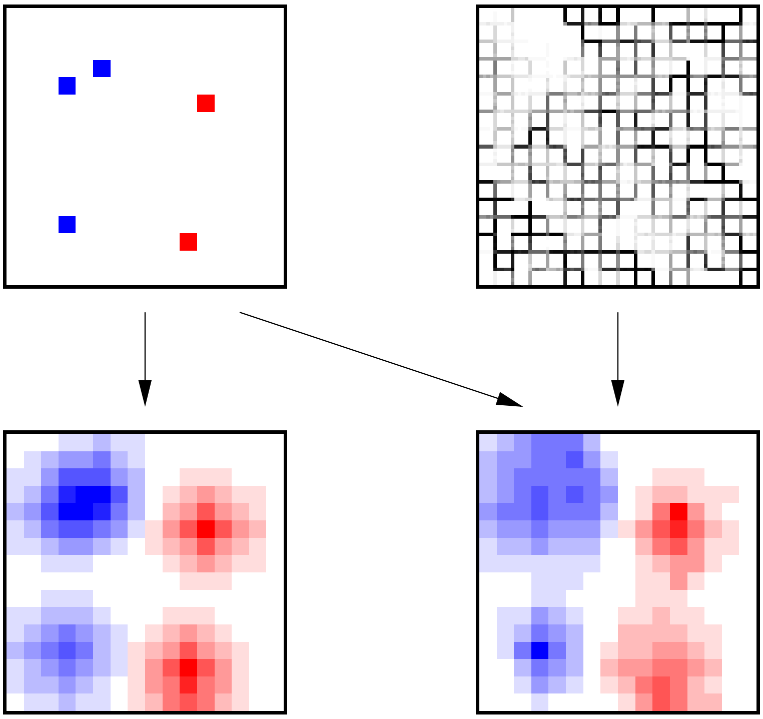

Figure 5.2: An example of how positive and negative map units, shown with red and blue marks on the top-left figure, are convolved. Shift-invariant convolution (bottom-left figure) is obtained with a fixed window function. Location-dependent convolution (bottom-right figure) takes also the relative distances between SOM units (top-right figure) into account. In the top-right figure, the relative distances are illustrated with gray level bars so that a darker shade of gray corresponds to a longer relative distance between neighboring map units.

a set of preliminary candidates $\mathcal{D}_m^\alpha,\ m = 1, 2, \ldots, M$, consisting of N_m^α images with the highest positive scores, is gathered for each of the M SOMs in use. The values of the parameters N_m^α have a clear effect on the computational requirements of the algorithm. To support the event that all images finally selected for showing to the user come from a single map, it should hold that $N_m^\alpha \geq N_n$. With large databases (N is large), the computational requirements often lead to $N_m^\alpha \ll N$.

For determining the final score or *qualification value* of an image, there exist now three alternate options. First of all, we can choose to disregard the possibility that an image which obtained a strong positive score on one SOM (and therefore appears as a good candidate image) obtained a strong negative score on another at the same time. In this case, duplicate entries for a single image in the subsets $\mathcal{D}_m^\alpha$ are simply removed so that the entry with the maximum value of the available scores is used as the qualification value of that image. This method also omits images which obtained moderately good responses from multiple maps but not strong positive responses on any map from the final set of images to be shown to the user. Second, we can take the above kind of situations into account and implement a stage of value combination for the images in the candidate sets. This can easily be done if we limit

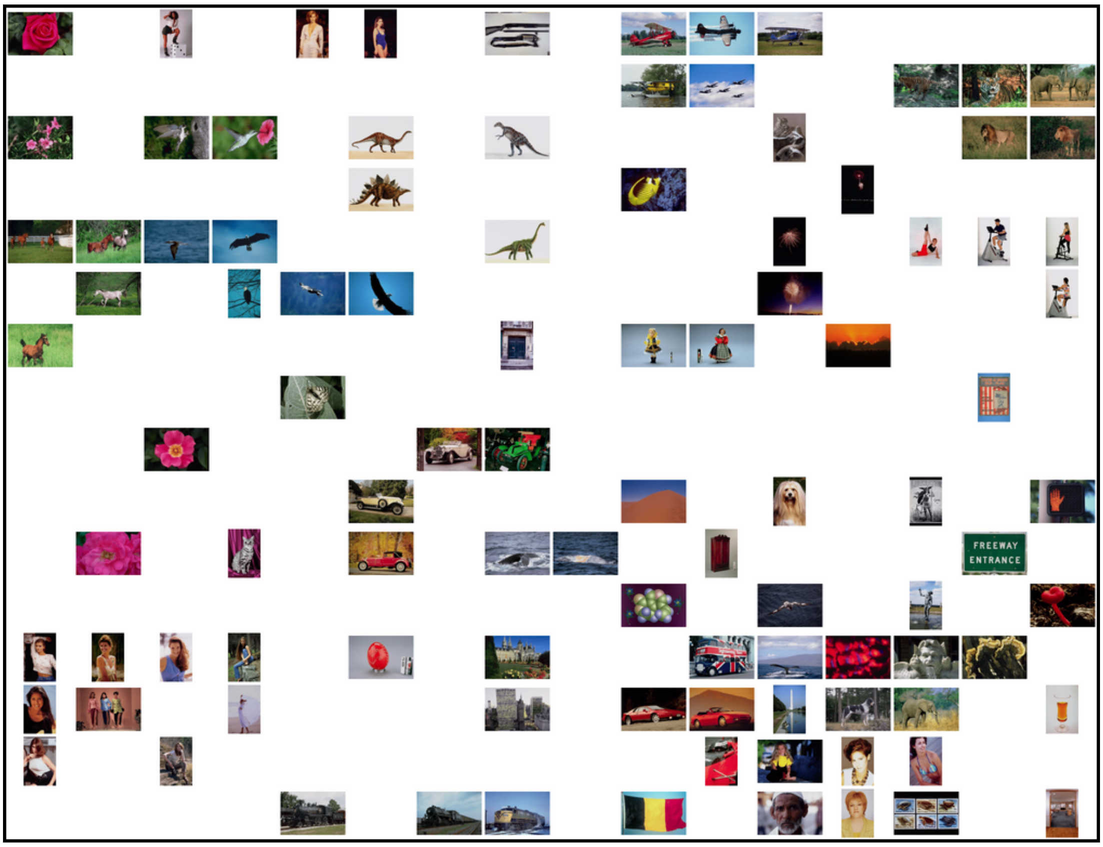

Figure 5.3: The image labels of a 16×16-sized SOM trained with user interaction data.

our consideration to the image instances present in the feature-wise sets as the scores are commensurable: we can, for example, simply sum the multiple scores together. The above two methods were compared in Publication IV, where they yielded rather similar results. The value summation method was used also in the experiments of Publications I and III. These methods are computationally advantageous as they only require a *sorted access* to the best-scoring images in the feature-wise image indexes to determine the sets $\mathcal{D}^{\alpha}_{m}$. The third option is to thoroughly consider all images present in the sets of preliminary candidates. For this purpose, we have to locate all these images in all the used SOMs and record all their feature-wise scores. These scores are then e.g. summed together to obtain the final qualification values. This method inevitably increases the needed processing in every query iteration as a *random access* to the mth SOM index is required to locate a candidate image if the image is not in $\mathcal{D}^{\alpha}_{m}$. On the other hand, this method maximally reinforces the interplay of the different features. It has been used in Publications V–VII.

The image retrieval method presented in this section conforms to the block diagram of a general CBIR structure presented in Figure 4.2. In the first stage, the feature-wise image subsets $\mathcal{D}^{\alpha}_{m}$ are gathered by determining the images with strongest positive relevance scores according to that particular feature. The combined set is then the union of the subsets, $\mathcal{D}^{\beta} = \bigcup_{m=1}^{M} \mathcal{D}^{\alpha}_{m}$. The second stage of processing consists of determining the final qualification values as described above.

The way the relevance feedback is implemented in this method has one additional advantage to be noted. As the cumulative responses are calculated for each SOM separately and the topologies of the feature spaces of the SOMs are different, the images which become selected due to the good performance of only one feature type are likely to be mapped in nonadjacent and sparsely distributed areas on the other SOMs. These areas will be neglected as long as areas with strong positive responses remain and contain unseen images. But, when good candidate images are no longer found, the search will proceed to these new regions. If more images from the neighboring map units are then marked as relevant by the user, these new areas of relevance will be discovered. The search will thus not be stuck in the local environments of the first relevant images found but will eventually expand to all neighborhoods of the different feature types of all positive seen images. This is important for two reasons. First, due to the possible folding effect of the SOM, in which the map folds in the high-dimensional feature space so that one original data cluster is represented by SOM units in more than one location of the map. Second, because the class of relevant images may form complex multimodal distributions in feature spaces due to the semantic gap.

5.4.5 Depth-first or breadth-first search

The policy of selecting the N_n best-scoring images as the ones to be shown to the user is valid when one or more areas of distinct positive response have been discovered. Concentrating the search on these areas, leading to a *depth-first type* search on the SOM structure, is justified as it can be assumed that the probability of finding more relevant images is high on these parts of the SOM grid. If this is not the case, it often is a better strategy to widen the scope of the search so that the user obtains a broader view of the database. For this purpose, we can use the mutual dissimilarity of the images as an alternative or secondary criterion. This leads to a *breadth-first type* selection of images. Breadth-first search can be directly implemented using the SOMs and their image labels. The image label of a SOM unit is the image whose feature vector is nearest to the model vector of the map unit (Section 3.3.1). Therefore, it can be considered as a kind of average among the images mapped to that map unit. Breadth-first search can thus be implemented by returning only label images of map units on the intermediate (i.e. non-bottommost) SOM levels.

An important use for breadth-first search is in the beginning of the queries if no initial reference images are available (the page zero problem, Section 4.1). In this mode of operation, it is important to return diverse images from the database. With TS-SOM structures, a natural compromise is to use depth-first search on the bottommost TS-SOM levels and breadth-first search on the other levels. Upper (i.e. smaller) TS-SOM levels generally have sharper convolution masks, so the system tends to return images from the upper TS-SOM levels in the early rounds of the image query. Later, as the convolutions begin to produce large positive values also on lower map levels, the images on these levels are shown to the user. The images are thus gradually picked more and more from the lower map levels as the query is continued. The balance between breadth-first and depth-first searches can be tuned with the selections of the size of the bottommost SOM level relative to the size of the image database and the window functions on different SOM levels. Initial reference images are not used and, therefore, this kind of setting is applied in the experiments of Publications I, III, and IV. On the other hand, if one or more initial reference images are available, there is no need for an initial browsing phase. Instead, the retrieval can be initiated straight in the neighborhoods of the reference image on the bottommost SOM levels as they provide the most detailed resolution. The upper TS-SOM hierarchy is thus neglected and only the largest SOMs of each feature are used. Experiments reported in Publications V–VII have been performed in this mode of operation.

5.4.6 Relations to other methods

The SOM can be seen as a method for clustering and for dimensionality reduction. The mapping of feature vectors and their associated images to the BMUs after the training of the map can be interpreted as clustering. This, however, ignores the topology of the SOM, so a portion of the provided data organization is dismissed. In most of the applications of the SOM in CBIR listed in Section 3.3.1, it is used to perform clustering. The SOM can be seen as a special case of vector quantization in which the neighborhood function (3.2) is not a delta function. The reference method used to compare PicSOM with in Publications III–V is directly based on this correspondence. The method is based on clustering the images by using the well-known k-means or Linde-Buzo-Gray (LBG) vector quantization (Linde et al. 1980). As reported in Publication IV, the LBG codebook yields better performance than the SOM used as a pure vector quantizer. This is understandable as the SOM algorithm can be regarded as a trade-off between two objectives, namely clustering and topological ordering. This trade-off is dependent on the size of the SOM; the clustering property is dominant with relatively small SOMs whereas the topology of the map becomes more significant as the size of the SOM is increased. Vector quantization has also been used in CBIR by other researchers, e.g. by Chen et al. (1997), Wood et al. (1998), Iyengar and Lippman (1998), Lu and Teng (1999), Yoo et al. (2002), Qiu (2002), and Ye and Xu (2003).

On the other hand, the SOM attempts to represent the data with optimal accuracy in a lower-dimensional space of fixed grid of points, thus performing dimensionality reduction. This functionality is integral to our relevance feedback method as the computational complexity of measuring image similarity is drastically reduced by transforming the original high-dimensional space into a two-dimensional grid. This makes our method scale well to large databases. For example, in Laakso et al. (2001), a 1024×1024-unit SOM was trained for a database of over 1 000 000 images obtained from the WWW, and the retrieval experiments in the publications included in this thesis have been performed using a database of 59 995 images. In the reduced space, the construction of sparse value fields and performing the low-pass filtering bear a similarity to using Parzen windows for density estimation (as was done in Meilhac and Nastar 1999). In addition to the dimensionality reduction and the SOM grid structure, another difference is that, in the PicSOM method, the result is interpreted only as a response invoked by the positive and negative examples and not as a probability density estimation of the class of relevant images. Therefore, negative cumulative responses do not matter and we can use the negative examples directly.

5.5 Long-term learning from relevance feedback

As discussed in this section, relevance feedback is a widely-used method for query improvement or *intra-query* learning. Current relevance feedback systems are generally designed so that the accumulated relevance information is discarded between successive queries. Each retrieval session is started from the same initial situation and preceding uses of the system have no influence on the present query. This is because the object of the search usually changes from one query to the next, and so the previous relevance assessments have no significance any more. Therefore, with relevance feedback, the learning is by nature intra-query, i.e. it takes place during a single query instance and the results are erased when beginning a new query.

As an additional property, the user–system interaction taking place in relevance feedback can be recorded and used in a *long-term* or *inter-query* learning scheme. Assuming binary relevance assessments, the feedback provided by the user during a specific query session divides the set of seen images $\mathcal{D}(n)$ into relevant images $\mathcal{D}^+(n)$ and non-relevant images $\mathcal{D}^-(n)$. These classes can be seen as subsets of the actual sets of relevant and non-relevant images ($\mathcal{D}^\oplus$ and $\mathcal{D}^\ominus$) with respect to the current query target. The fact that two images belong to the same relevance class is a cue for similarities in their semantic contents. On the other hand, using previously stored retrieval sessions, presumably performed also by other users of the system, might conflict with the subjectivity and context-dependency of human notion of image similarity. In practice, however, it turns out that previous user assessments provide valuable accumulated information about image semantics and can be a considerable asset in improving retrieval performance, albeit being static in nature.

5.5.1 Existing approaches to long-term learning

While intra-query learning by relevance feedback has achieved prevailing popularity in current CBIR, less research has been focused on exploiting long-term learning. However, a number of approaches have recently been presented and incorporated into various CBIR systems. In *FourEyes*, the image groupings and their associated weights are stored across query sessions and updated based on the new user interaction information (Minka and Picard 1997). In *MetaSeek*, all user interactions were stored and used in later queries in selecting between a set of independent image search engines (Benitez et al. 1998). Müller et al. (2000b) presented a method where the log files of their *Viper* system (later renamed to *GIFT* or *GNU Image Finding Tool*) are used to adjust weights for different feature components. A Bayesian framework for both short-time and long-time learning was presented in Vasconcelos and Lippman (2000). Graph-based methods have been presented by Sull et al. (2000) and Zhang and Chen (2002). The images in the database are represented as nodes and the semantic similarity between two images as an arc between the corresponding nodes. Heisterkamp (2002) presented the idea of using VSM and LSI by considering the images as the vocabulary of the system and the classes of relevant images as documents whose words are the images. Li et al. (2002b) presented a method where feature-based similarities are first computed and the images are ranked correspondingly. Then, the images are re-ranked based on pair-wise semantic correlations obtained from recorded relevance feedback. In Fournier and Cord (2002), the visual similarity measure is weighted by a long-term similarity measure.

5.5.2 User interaction feature in PicSOM

In most approaches to utilizing relevance feedback in long-term learning, the recorded user interaction information is used to weight or otherwise modify the results the system would normally return with the existing visual features. These approaches are valid and lead to improved results as reported in the research articles cited in the previous section. Still, an alternative approach is to consider the previous user interaction as metadata associated with the images and use it to construct a *user interaction* or *relevance feature*, to be used alongside with the visual features. In the PicSOM framework, this approach has desirable properties since one of the strengths of the system is that it inherently uses multiple features and generally benefits from adding new ones. This way the user interaction data is treated similarly as any other source of information about image similarity without the need for any special processing. This method was presented and experimented with by the author of this thesis in Publication VII.

As already discussed, the user evaluates the shown images either as relevant or non-relevant during a query with PicSOM. In the context of long-term learning, we only consider the set of relevant images gathered during the query. It is assumed that since these images were all selected as relevant during a single query, they share common semantic characteristics. This information is coded into a binary feature which has the value one for the images in the set of relevant images and zero for other images. A finer granularity of relevance assessments could also be supported by converting the assessments to suitable scalar values.

The basis for the user interaction feature is the vector space model of textual documents (Section 5.2.1) where the m documents in a corpus are represented by the words in them by using an $n \times m$ term-by-document matrix $\mathbf{X}$, where n is the number of different words. The dimensionality of $\mathbf{X}$ is then reduced by LSI, i.e. first applying singular value decomposition:

$$\mathbf{X} = \mathbf{U}\,\mathbf{S}\,\mathbf{V}^T \tag{5.16}$$

where $\mathbf{U}$ and $\mathbf{V}$ are $n \times r$ and $m \times r$ orthonormal matrices, and $\mathbf{S}$ is an $r \times r$ diagonal matrix containing the singular values of $\mathbf{X}$ on the diagonal and $r \leq \min(n, m)$ is the rank of $\mathbf{X}$. After the decomposition, we only consider k ($k < r$) dimensions corresponding to the k largest singular values of $\mathbf{S}$:

$$\widehat{\mathbf{X}} = \widehat{\mathbf{U}}\,\widehat{\mathbf{S}}\,\widehat{\mathbf{V}}^T \approx \mathbf{X} \tag{5.17}$$

where $\widehat{\mathbf{S}}$ is a $k \times k$ diagonal matrix containing the k largest singular values and $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{V}}$ are $n \times k$ and $m \times k$ matrices containing the corresponding left and right singular vectors, respectively. Thus, we obtain a representation of the originally m-dimensional data in k dimensions as the rows of $\mathbf{Y} = \widehat{\mathbf{U}}\,\widehat{\mathbf{S}}$.

In this setting, VSM is applied as in Heisterkamp (2002). That is, instead of considering images as documents as in the retrieval phase, here the user-provided relevance evaluations are considered as the documents (i.e. one recorded query equals one document) and the images in the database as the words in the vocabulary. Term or image frequency weighting is unnecessary as each image may appear at most once in one relevance evaluation, but document frequency weighting, i.e. weighting elements of $\mathbf{X}$ by the number of documents in which the corresponding term occurs in, can be applied. In our setting, LSI is primarily used to perform dimensionality reduction. This is needed as the dimensionality of the data equals the number of image queries in the training data, which may well be in the order of hundreds or thousands and thus excessive for direct usage in SOM training.

The rows of matrix $\mathbf{Y}$, each corresponding to one image, are treated as a user interaction feature of dimensionality k and the corresponding TS-SOM is trained and used in parallel and similarly as the TS-SOMs trained with visual features. An example of a resulting SOM is illustrated in Figure 5.3. In the figure, the 16×16-sized map level of a TS-SOM trained with user interaction data, gathered as described in Publication VII, is shown. It can be observed that images with similar semantic content have been mapped near each other on the map. The sparsity of the map is a direct consequence of the sparsity of the data; the cardinality of the set of relevant images is typically much lower than the size of the database and the images in the same relevance evaluation tend to get mapped into the same map unit.

5.5.3 Hidden annotations

In some cases, the image database may contain manually assigned or implicit annotations as discussed in Section 3.1.2. These annotations describe high-level semantic content of the image and often contain invaluable information for retrieval. Therefore, it is useful to note that the user-provided relevance evaluations discussed above are notably similar to hidden annotations (Section 3.1.2). In particular, hidden annotations can be seen as high-quality user assessments. Images having a certain term in their annotations can be seen as the set of relevant images when the user was querying for images containing the concept corresponding to the term in question. Alternatively, hidden annotations can be regarded as a goal toward which the user interaction feature evolves as more user interaction data is recorded. The technique described in the previous section can thus be readily utilized for keyword annotations, even if only a subset of the images contains these annotations. This was also experimented with in Publication VII.

6 RETRIEVAL EVALUATION

Evaluating CBIR systems is paramount for further development in the research field. A wide range of retrieval techniques and feature extraction methods have already been proposed and it is essential to be able to objectively compare these in order to identify the most efficient ones for various image retrieval tasks and domains. This would guide research into right directions and lead to improvements in used techniques and overall development of better image retrieval systems. For a specific CBIR system, it is also useful to be able to evaluate the effects of changing environment: does an established method continue to perform well when e.g. the image domain or the query type is changed.

Researchers in text-based IR have long identified retrieval evaluation as a challenging and important problem and addressed it for decades. The inherent subjectiveness associated with deciding the relevance of a document, which in fact can be seen to distinguish information from data and IR from ordinary data processing, makes objective performance evaluation difficult. Early research was carried out already in the 1950s and, in the 1960s, a seminal effort on the field was the *SMART* system (Salton 1971). A major milestone was accomplished when the annual Text REtrieval Conference (TREC) project was started in the early 1990s (TREC 2003). TREC provides a common benchmark setup with large test collections. For each conference, a set of reference experiments is designed and participants use these experiments for comparing their IR systems. Different aspects of IR are highlighted on various tracks, e.g. user interaction is studied on the interactive track. Since 2001, TREC has also included a benchmark for visual data in the video track (Smeaton and Over 2003, TRECVID 2003). Overviews of retrieval evaluation in IR are presented e.g. in Salton and McGill (1983), Salton (1992), and Baeza-Yates and Ribeiro-Neto (1999). Many of the well-established solutions in IR can directly be used in CBIR, so the IR methodology provides a natural starting point for CBIR system evaluation.

Human subjectivity plays perhaps an even more prominent role when dealing with visual data. As each user of an image retrieval system has her individual expectations and an image can be relevant to a query in a multitude of ways, there does not exist a definite right answer to an image query. User assessments of images often vary considerably, as was shown e.g. in the experiments performed by Squire and Pun (1998). Still, some kind of a ground truth classification of images is usually performed to form a basis for the evaluation process. A simple method—employed also in the experiments performed in the publications included in this thesis—is to form a set of image classes by first selecting appropriate verbal criteria for membership in a class and then manually assigning the corresponding Boolean membership value for each image in the database. In this manner, a set of ground truth image classes, not necessary non-overlapping, can be formed and then used as the basis of retrieval evaluation. Unfortunately, obtaining these relevance judgments may be difficult and costly, particularly in certain specialized domains, e.g. with medical images, where making these assessments requires domain expert knowledge. Furthermore, this approach requires an exhaustive examination of the whole database, which may be infeasible with large databases and makes it evident that there will be a perpetual lack of ground truth information even if suitable image databases would be abundantly available for CBIR researchers. In TREC, the following *pooling* technique is used instead. First, a pool of possibly relevant documents is obtained by gathering rather large sets of documents (e.g. 200 as in TREC-1) returned by the participating retrieval systems. These sets are then merged, duplicate documents are removed, and the relevance of only this subset of documents is assessed manually (Harman 1992). The set of relevant images can also be obtained by transforming the query image, for example, by rotation, scaling, cropping, or adding noise (Lew and Denteneer 2001, Li et al. 2002a).

Still, a more pressing issue is the lack of common and standardized benchmarks. CBIR researchers use different image collections, query images, retrieval settings, and evaluation measures when reporting on the performance of their methods. Tuning the settings of the evaluation environment to highlight the advantages of the presented method is tempting, since retrieval algorithms often have distinct strengths and weaknesses. Unfortunately, this makes it essentially impossible to compare the relative superiority of different methods without standardized evaluation benchmarks. An informative discussion on the relativity of CBIR benchmark results is presented in Müller et al. (2002), where it is shown that even with a single source of images (Corel Photo CDs), it is possible to obtain almost arbitrary results by adjusting the evaluation parameters. Therefore, the need for a standardized full evaluation suite is persistent.

The ultimate measure of CBIR system performance is the satisfaction of the system's users. Therefore, experiments with human users will become inevitable at some point of system development. This kind of experiments are, however, laborious and time-consuming as they require a lot of human effort and test users. In addition, these evaluations are by nature subjective, so extreme care is required for performing system comparisons with test users. User judgments of the merits of one retrieval system over another depend on multiple causes, including the subjective overall ease of achieving the desired result, responsiveness of the system, whether the system matches the preconceptions and intuition of the user, and the user interface in general. Typically, these issues have not been much contemplated in research prototype systems. For these reasons, it would be advantageous to be able to automate the evaluation process.

For one-shot queries with QBE and ground-truth relevance classes this is relatively straightforward, as we can use images for which the ground-truth class is known as reference images and measure how well the system is able to return images belonging to the same ground-truth class. This approach can be applied to both target and categorical searches. For systems using relevance feedback, the task is more complex as the system has to be able to decide the relevance scores of images returned during a query session. A common solution is to use categorical feedback (cf. Section 5.1) although it may not be the optimal query strategy. An experienced human user might obtain better results with more flexible relevance assessments and possibly backtracking in the query structure if the results seem to deteriorate.

6.1 What to evaluate?

Individual features can be studied separately and independently from the other features for assessing their capability to efficiently index visual content. Such an analysis should account both for local and global clustering of image classes since semantic image classes may form complex multimodal densities in the feature spaces. In this type of evaluations, it is often unnecessary to simulate the actual retrieval system, which can be time-consuming with an extensive evaluation set-up. Rather, a direct measure based on the ability of the feature extraction to discriminate images belonging to a certain set of semantic similarity or relevancy from other images may be sufficient. Within the PicSOM project, this kind of experiments have been performed in Publication I and also e.g. in Oja et al. (1999), Brandt et al. (2002), and Laaksonen et al. (2003). Other research groups have also reported many studies on the performance of separate features in different image retrieval settings; see e.g. Ma and Zhang (1998), Di Lecce and Guerriero (1999), Rubner et al. (2001), or Ojala et al. (2002). Evaluations of this type are essential in developing effective feature extraction methods for CBIR.

For CBIR system development, the feature-wise assessments, however, have limited usefulness as they do not generally portray the operation of the entire CBIR system. Often, an effective CBIR system has to rely on multiple features, as was discussed in Section 4.2.2. In addition, a straightforward feature-wise evaluation does not take any relevance feedback mechanism into account. Therefore, there is a clear need also for retrieval efficiency evaluations of whole CBIR systems. A broad set of this kind of experiments are presented in Publications I and III–VII. The block structure approach for CBIR systems, discussed in Section 4.4 and Publication IV, can also be utilized here. If the retrieval system has been built as a series of smaller functional blocks, we can assess the retrieval performance of different paths through the block structure and discover the one with the best performance in the given setting, as was done in Publication IV.

In addition to the efficiency of the retrieval, other measures for evaluating CBIR systems should also be considered. An important criterion is time, as an interactive system has to promptly present the results. So far this has been deemed secondary and the focus has been on evaluating performance by the retrieval results only. Anyhow, when designing real CBIR applications, time requirements have to be taken into account, presumably resulting in inevitable trade-offs between speed and efficiency of retrieval. A related evaluation criterion is the scalability of the retrieval method, especially with respect to the size of the database, N. Although the database size may be small in some specific settings, general-purpose CBIR systems have to be able to handle large numbers of images without severe performance deterioration or excessive storage space requirements for the image indices. Generally, this means that the time complexity of the algorithm should be sublinear and the space complexity polynomial, preferably linear, with respect to N. Evaluation benchmarks should therefore be designed to use sufficiently large data sets. For example, in Gunther and Beretta (2000), it was suggested that even an initial benchmark database should contain at least 10 000 images. The scalability requirement has also been long recognized in the IR community and TREC, for example, has always concentrated on retrieval from large test collections. Other relevant criteria for CBIR system evaluation include the flexibility of the system to different application areas and environments, robustness, and user interface issues, etc.

6.2 Evaluation measures

The best-known and most widely used measures of retrieval efficiency in IR are *precision*

$$\mathcal{P} = \frac{\textit{number of relevant items retrieved}}{\textit{total number of retrieved items}} \tag{6.1}$$

and *recall*

$$\mathcal{R} = \frac{\textit{number of relevant items retrieved}}{\textit{total number of relevant items}} \ . \tag{6.2}$$

To some extent, these two measures are opposed to each other as precision is usually higher in the beginning of a query and it deteriorates as more items are returned. On the other hand, if the whole database is returned, recall reaches one, but precision is low i.e. the *a priori* probability of relevant items. As a result, both precision and recall are insufficient measures when used alone and should either be used together, e.g. precision when recall attains a certain value, or at a fixed *cutoff* point, i.e. when a fixed number of database items have been returned. Precision and recall are also commonly represented as a *recall-precision graph*, in which precision values are plotted against values of recall. Both precision and recall can also be plotted against the number of retrieved images. These graphs are very informative methods for illustrating system performance and e.g. show clearly the effect of relevance feedback. Comparing two graphs is, however, more difficult than comparing scalars, so interpreting recall-precision graphs requires some experience. In the experiments performed in the publications included in this thesis, the recall-precision graph was used as the performance evaluation measure in Publications V and VII. Still, due to the difficulties in dealing with two evaluation parameters, several methods for combining precision and recall to a single measure have been proposed, although none of these measures contain as much information as recall-precision graphs. For example, *average precision* is obtained by computing precision at each point when a relevant item is found and then averaging these precision values. With the *F-measure*, a parameter $\alpha \in [0, 1]$ is used to attach degrees of importance to precision and recall and a single measure is obtained with

$$F = [\alpha(1/\mathcal{P}) - (1-\alpha)(1/\mathcal{R})]^{-1} \ . \tag{6.3}$$

Another commonly used approach to retrieval performance measurement is to use some of the existing rank-based methods. The *rank* of an image is defined as its ordinal number among the returned images. For target search, a simple measure is the rank of the target image. In category search, the ranks of relevant images are usually averaged or, in some cases, the rank of the first relevant image is used. Variations of the average rank include *normalized average rank* (Müller et al. 2001), the *BIRDS-I measure* (Gunther and Beretta 2000) and the so-called τ *measure* used by our research group (Publications I, III, IV, and VI). The τ measure coincides with the question "how large portion of the whole database needs to be browsed through until, on the average, the searched image will be found" when we assume that one (random) image from the class in question is the actual target image the user is looking for and she uses categorical feedback to guide the retrieval system. Since the selection procedure of relevant images is fixed, this process can be automated. To eliminate the effect of different initial configurations, a common setup is to provide a single initial example image to start the query with. A complete treatment of an image class is then obtained by using every image in the class one at a time as the example image. The τ measure is obtained by calculating the average ranks of all relevant images and dividing their average with the size of the database. It yields a value in the range $\tau \in [\frac{\rho}{2}, 1 - \frac{\rho}{2}]$ where ρ is the *a priori* probability of the ground-truth image class in question. For values $\tau < 0.5$, the performance of the system is thus better than random browsing of images and, in general, the smaller the τ value the better the performance.

As discussed above, retrieval performance can be evaluated with many different measures and any single measure cannot capture all aspects of the retrieval performance of CBIR systems. Therefore, it may be justified to use an extensive repertoire of measures, especially when comparing two considerably different systems. Recommendations for selections of benchmark measures are provided by Smith (1998), Leung and Ip (2000), and Müller et al. (2001).

6.3 Standardization initiatives

Performance evaluation in CBIR is currently more or less in a state of disarray and far from the level of maturity of retrieval evaluation in IR. However, the need for benchmark standardization has been recognized in the field, and notable initiatives have been started. The first objective is to gather common image collections, ground truth relevance classes, and test queries, which would be freely available to all research groups. The *de facto* standard image collection has been to use images from the Corel Photo CDs, which usually contain sets of 100 images each under a common theme. Unfortunately, there is no single uniform Corel image set and thus the Corel databases different research groups possess are usually not identical. This leaves room for tuning the database in order to obtain desirable results, as illustrated in Müller et al. (2002). In addition, the Corel images are copyrighted and no longer even available. Therefore, researchers have begun to compile their own royalty-free collections. These include the University of Washington database

(ANN 2003), the Benchathlon database (Benchathlon 2003), and the IAPR Technical Committee TC12 database (TC12 2003). At the moment, these databases are rather small, containing only a few thousand images, and not fixed. As yet, none of them has obtained wider endorsement. In addition to a common database, a complete benchmark setting would also require a representative set of ground truth classes and test queries. Especially the compilation of ground truth classes is difficult and time-consuming. Commercial image databases, for example the Corel Photo CDs, occasionally contain manually constructed annotations, which can be used as ground truth, as e.g. in Barnard and Shirahatti (2003), although these annotations may be somewhat inconsistent with the image content. Therefore, for our experiments with the Corel images, we created our ground truth image classes manually with clearly specified membership criteria (see Section 6.4).

MPEG-7. MPEG-7 (MPEG 2002, Manjunath et al. 2002), formally "Multimedia Content Description Interface", is an ISO standard developed by the Moving Pictures Expert Group. MPEG-7 aims at standardizing the description of multimedia content data. Among other issues, it defines a standard set of Descriptors that can be used to describe various types of multimedia information. One of the main application areas of MPEG-7 will undoubtedly be to extend the current modest search capabilities for multimedia data for creating effective digital libraries. It is expected that MPEG-7 will have a similar prominent impact on multimedia content description as the previous MPEG standards (MPEG-1, MPEG-2, and MPEG-4) on their respective application areas.

Color Descriptors	**Texture Descriptors**	**Shape Descriptors**
Dominant Color	Edge Histogram	Region-Based Shape
Scalable Color	Homogenous Texture	Contour-Based Shape
Color Layout	Texture Browsing	Shape 3D
Color Structure		
GoF/GoP Color		

Table 6.1: MPEG-7 Visual Descriptors applicable for still images.

As a non-normative part of the standard, a software eXperimentation Model (XM) (MPEG 2001, MPEG 2003) has been released for public use. The XM software is a framework for the reference code of the standard. In the scope of this work, the most relevant part of XM is the implementation of MPEG-7-defined Descriptors. Table 6.1 lists MPEG-7's Visual Descriptors applicable for still images. From a CBIR benchmarking viewpoint, a set of common feature extraction methods is extremely useful as it can be used to remove the effect of different features from the evaluation. MPEG-7 Descriptors have been used as visual features in the experiments of Publications V–VII.

Benchathlon. Benchathlon (Benchathlon 2003) is an initiative for creating a public contest to assess the merits of various image retrieval algorithms. The aim of the initiative is to set up a collaborative environment where standard CBIR evaluation protocols and frameworks can be developed. The leading principle in designing the benchmark has been to use a distributed client–server architecture, as described in Gunther and Beretta (2000). The purpose is to divide CBIR systems into separate client and server parts, in order to be able to measure CBIR performance over the Internet. For the client–server communication, a specific language, Multimedia Retrieval Markup Language (MRML) (Müller et al. 2000c, MRML 2003), has been developed.

The support for MRML has recently been added also to the PicSOM system (Rummukainen 2003). As PicSOM includes a benchmarking tool which has been used to perform all the experiments with the system, we can now use it for testing other MRML-based CBIR systems as well. In anticipation of a general Benchathlon contest, a set of comparable experiments with PicSOM and *GIFT* (Squire et al. 1999a, Squire et al. 2000), which is both publicly available and uses MRML for client–server communication, are presented in Rummukainen et al. (2003) and more broadly in Rummukainen (2003).

Generating ground truth classes has been a major issue within the Benchathlon project. Building and testing a general vocabulary for image annotation is discussed in Jörgensen and Jörgensen (2002). Pfund and Marchand-Maillet (2002) presented an image annotation tool designed to aid in the tedious task of annotating large databases.

6.4 Summary of experiments

An extensive set of experiments was performed during the research project described in this thesis. Distinct experiments were carried out and described in Publications I and III–VII. In this section, an overview of the experiment settings and the obtained results is presented.

All experiments in this thesis have been performed using a database of 59 995 miscellaneous images originating from Corel Photo CDs (the Corel Gallery 1 000 000 product). The images are mainly photographs, most of them in color, but a small number of artificial images are also included. The database was originally compressed with a wavelet compression algorithm and was thus first locally converted to JPEG format with a utility provided by Corel. The size of each image is either 384×256 or 256×384 pixels. For other studies on using the PicSOM system with different databases, see Oja et al. (1999), Laakso et al. (2001), Viitaniemi and Laaksonen (2002), Iivarinen and Pakkanen (2002), or Matinmikko (2002).

A number of different image features have been extracted from the database images. The used features in the experiments of Publications I, III, and IV were implemented by our research group. These features are Average color, Color moments, Texture neighborhood, Shape histogram, and Shape FFT. Within the included publications, the most detailed description of these visual low-level features is presented in Publication II. A comprehensive treatment of the shape-based features is given in Brandt et al. (2002). Apart from Shape FFT, all the above features were calculated separately in five fixed image regions (see Publication II). In Publication V, a set of MPEG-7 Descriptors (Section 6.3) applicable for still images was adapted as features for the PicSOM system and these Descriptors were used as visual features in the experiments of Publications V–VII. The MPEG-7 Descriptors were always calculated from the entire image. In Publication VII, additional features based on recorded user interaction data and existing keyword annotations were presented and experimented with.

The images in the used database have been grouped by Corel into thematic groups, usually consisting of 100 images each. Keyword annotations for almost every image are also provided. However, we found these image groups and annotations often rather inconsistent with the content of the images. Therefore, we created for the experiments a total of seven manually-picked ground truth image sets with tight membership criteria. All image sets were gathered by a single subject. The used sets and membership criteria were:

- **faces**, 1115 images (*a priori* probability 1.85%), where the main target of the image has to be a human head which has both eyes visible and the head has to fill at least 1/9 of the image area.
- **cars**, 864 images (1.44%), where the main target of the image has to be a car, and at least one side of the car has to be completely shown in the image and its body to fill at least 1/9 of the image area.
- **planes**, 292 images (0.49%), where all airplane images have been accepted.
- **sunsets**, 663 images (1.11%), where the image has to contain a sunset with the sun clearly visible in the image.
- **houses**, 526 images (0.88%), where the main target of the image has to be a single house, not severely obstructed, and it has to fill at least 1/16 of the image area.
- **horses**, 486 images (0.81%), where the main target of the image has to be one or more horses, shown completely in the image.
- **traffic signs**, 123 images (0.21%), where the main target of the image is one or more official traffic signs, so commercial and other signs were rejected.

In order to make it feasible to run automated experiments, categorical feedback was assumed in all experiments. This basis enables the construction of justifiable test cases of target and categorical searching at query levels 1 and 2 (Section 2.2). More vague retrieval scenarios, such as level 3 queries or free image browsing, would undoubtedly require a more elaborate experiment setup. Two distinct cases were experimented with for the initial setting of the query. In Publications I, III, and IV, we assumed no initial reference images in the beginning of the query and the retrieval began with an implicit browsing phase. In Publications V–VII, the retrieval was initiated by providing one relevant example image. For details, see Section 5.4.5.

It may be argued that the scope of the performed experiments remains somewhat limited and further experiments e.g. with actual test users would be beneficial. However, a study in which the PicSOM system was compared with other retrieval systems in an experiment settings involving test users has been conducted by Matinmikko (2002). The database visualization and image browsing side of the PicSOM system was inevitably belittled due to the type of the performed experiments. User interface aspects have clearly remained secondary within the system and should be underscored in further development.

The results of the single-feature experiments in Publication I showed that the tested shape features outperfomed the simple color and texture features, which is understandable as they undoubtedly are more sophisticated. Consistent results were obtained by directly measuring the features' abilities to discriminate images belonging to a certain category and by using the actual retrieval system with a single feature.

From the experiments involving varying combinations of multiple features (Publications I and V), a prevailing observation is that the use of a larger set of features generally yields better results than using a smaller set. Most notably, the best results are usually obtained by utilizing all available features. Therefore, it can be stated that the proposed retrieval technique provides a robust method for using a set of different features in parallel. The importance of this observation should be emphasized, since general-purpose CBIR systems will undoubtedly have to rely on multiple features and the users do not generally possess the necessary background knowledge required for initial feature selection.

The proposed method was compared with a reference method based on vector quantization (VQ) in Publications III–V. The SOM turned out to be inferior to the k-means algorithm as a plain clustering algorithm. This is understandable as a considerable part of the indexing power of the SOM lies in the preservation of topology. When comparing k-means clustering and the proposed SOM indexing method with the τ performance measure, the observed results were rather similar to each other. In Publication III, the best results were obtained with k-means vector quantization. The recall-precision curves of Publication V, however, display considerably different retrieval behavior and illustrate the efficiency of relevance feedback with the proposed method. The initial precision of VQ is better but the ranking of the methods' precision values can switch even after one or two query rounds. Overall in the experiments of Publication V, the reported performance of the proposed method is increased due to a change in the retrieval algorithm. Now, all the images present in the feature-wise candidate sets are considered thoroughly in the second stage of processing (see Section 5.4.4). This increase of performance does translate into increased computational requirements, but the practical effect of this increase turned out to be rather insignificant. Moreover, the immensely useful ability to perform feature selection is not observed with the VQ method as adding inferior features soon begins to degrade retrieval results. Using scalar quantization (SQ) was also experimented with in Publication IV, but the results obtained with SQ were clearly inferior to the results of VQ or SOM-based methods.

In addition, the relative performances of MPEG-7's Color Descriptors, excluding the GoF/GoP Color, were experimented with in Publication V. The results indicate that none of the tested Descriptors seems to dominate the other Descriptors and the results vary considerably on different image classes. This result can be seen to emphasize the need to use many different features in parallel. Also, the results were similar regardless of the used retrieval algorithm (the proposed method or the VQ-based reference).

An informative by-product concerning the ground-truth image classes can be obtained by visualizing how the image classes are mapped on different SOM surfaces (Publications II, III, and V). With large SOMs, it is useful to low-pass filter the distributions in order to ease the inspection. This kind of visualization reveals the capability of a feature extraction method to map similar images near each other in the feature space and also the SOM training algorithm's ability to preserve the spatial ordering of the feature space. In general, the visual inspection of the obtained distributions conformed with the results obtained by running the retrieval algorithm.

Choosing the size and shape of the convolution window, which are central parameters of the presented SOM-based relevance feedback method, was studied in Publication VI. With the experiments, it was verified that using a small window length is sufficient as the best results were obtained with windows of size $2 \leq l \leq 4$ (see Section 5.4.2). This result is valuable especially when considering computational requirements of the online part of the method. Apart from the poorly-working rectangular window, the exact shape of the window function was found to be rather insignificant. In addition, using location-dependent window functions was also experimented with in Publication VI. In the used experiment setting, they did not improve the results, but may still prove out to be useful in some applications.

In Publication VII, a method for long-term learning based on previous user–system interaction was presented and experimented with. Based on the experiments, it was observed that the recorded usage data can improve retrieval precision considerably, even with only a rather small number of queries available. This enables the development of retrieval systems which increase their performance gradually as they are used. For semantic image retrieval, this can be a considerable asset, especially if no actual semantic annotations are available. The presented method was also tested with real keyword annotations. Using these can lead to greatly improved precision especially in category search, as was observed in the experiments. Overall, the best results were nonetheless obtained by incorporating the visual features with the keyword feature.

7 CONCLUSIONS

Image retrieval is a lucid example of user-centric computing. Image relevance or semantics cannot be objectively defined and the correct action of a retrieval system is always context and user-dependent. This makes designing and especially evaluating automatic tools for CBIR a challenging task.

On the other hand, a distinct characterizing aspect of CBIR is the semantic gap. Potential users of these technologies are generally interested in searching for images of particular semantic attributes, scenes, objects, or events and the similarity measurements based on low-level features are not always able to provide this functionality. Therefore, with general images, additional data may be needed for reaching an acceptable performance level.

Thirdly, emphasis in CBIR research should be increasingly placed on the effectiveness of the developed techniques. Visual data is evermore widespread and image databases will be required to store and handle massive amounts of data. Applications of this kind will also be more and more designed for handheld and mobile devices with limited processing and memory resources instead of standard desktop workstations and servers. Scalability should, therefore, be a major concern when developing general-purpose techniques for CBIR.

At present, CBIR has been an active research topic roughly for a decade. Although a number of prominent advances have undoubtedly been achieved, none of the current systems has established wide success or adoption, and practical image retrieval is still mostly based on manually entered or implicit textual annotations. This should not be regarded as discouraging due to the evident difficulties in developing CBIR systems with a level of sophistication comparable to the current state-of-the-art in traditional IR. Rather, CBIR should be seen as a rapidly evolving but highly challenging research topic. Note that even text-based IR remains a widely researched issue after at least forty years of significant research effort.

The literature survey and the bibliography in this work aim at presenting a summary of notable recent work and advances in the field of image indexing and retrieval. An all-inclusive treatment of relevant research to the topic is no longer feasible nor was the intent here due to the vast amount of novel works published in the recent years. Still, certain methodologies can be seen to have achieved a considerable level of maturity and also recognition in the field. An effort was made to bring these forward in this introductory part of the thesis. All the presented types of indexing methods, viz. dimensionality reduction, recursive partitioning, clustering, vantage points, and inverted file, have their characteristic strengths and weaknesses. Therefore, the distinct criteria of the application in question should dictate the selection of the indexing method. The same applies also to different methods for relevance feedback presented in this work. With the relevance feedback methods, however, the suitability for general retrieval applications varies considerably due to diverse computational requirements. Apart from classical methods originating from IR, such as query point movement and feature component re-weighting, the techniques are rather immature. Considerable improvements can thus still be expected.

The central part of this thesis is a novel CBIR system framework, the PicSOM system, developed by the author and his coworkers. The leading principle has been to develop a general system structure and to study effective relevance feedback techniques suited for SOM-based image indices. Image indexing with the SOM was perceived to be a robust and effective solution which tolerates even very high input vector dimensionalities. As an indexing method, the SOM was interpreted as a combination of clustering and dimensionality reduction. Unlike basic clustering algorithms, the SOM has the advantage of providing a natural ordering for the clusters due to the preserved topology. This way, the relevance information obtained from the user can be spread to neighboring image clusters. The dimensionality reduction aspect of the algorithm alleviates computational requirements of the algorithm; for time-critical online processing the high-dimensional feature space is substituted by a two-dimensional grid. Dimensionality reduction to 2D also enables straightforward image database visualization, for which the rectangular SOM grid is inherently suited.

One drawback is that the SOM algorithm is linked with the Euclidean distance measure in its basic form. Euclidean distance was thus used with all features in this work. For certain features this probably is suboptimal as for example the MPEG-7 Descriptors have their own distance measures defined. However, with experiments it was observed that the Euclidean SOM yields a serviceable index also in these cases. Specific treatment for each feature separately could result in performance increase, although this would require a more complex training algorithm. Still, the online part of the presented retrieval method would remain unaltered as the method for constructing the SOM indices has no bearing on the relevance feedback technique.

A number of divergent experiments were performed in the included publications, demonstrating the versatility of the proposed technique. Overall, the results establish that relevance feedback with the proposed method is indeed able to adapt to different query types. Unlike many existing relevance feedback methods, the proposed method can also take negative examples into account in a straightforward manner. In the experiments, a broad set of different features were employed. Among these were a subset of visual descriptors from the MPEG-7 standard and features based on previous user interaction and keyword annotations. These diverse features are all represented in a common way, i.e. with a SOM structure.

Scalability to large databases has invariably been a central design principle in the development of the PicSOM retrieval method. Scalability of the method was verified with experiments reported in the included publications, which were performed with a large image database of 59 995 images. In addition, we have successfully indexed also a database of over a million images from the WWW with the PicSOM indexing method. With suitable values to a few parameters, such as the convolution window length and the number of images considered in the intermediate stages, which control the computational complexity of the online processing, the presented method is capable of response times of only a few seconds even with large image databases. Another important parameter is naturally the size of the bottommost TS-SOM level as it controls the interplay of clustering and topological ordering tendencies of the indexing method. Large SOMs have a more detailed resolution and can be beneficial in many application domains. By using the TS-SOM algorithm, the computational requirements of training large maps is drastically reduced.

The state of the retrieval system described in this thesis leaves a lot of interesting directions for further research. In this work, the focus has been toward universality of the retrieval method. According to this point of view, the discussion has been intended to be neutral to any application domain, database type, or the repertoire of available features. In specific application domains, the general approach may not be optimal and domain-specific development and modifications are likely to be justified.

Although much work has been done in developing efficient feature extraction methods for CBIR, there is still room for considerable improvement. The MPEG-7 descriptors can be regarded as the current baseline, against which new feature extraction methods can be compared. However, the most prominent advances are likely to be achieved with further research on intermediate semantic features.

Image database visualization and inexact forms of image retrieval are the flip side of typical CBIR embodied also in the relevance feedback technique proposed in this thesis. The user interface of the system does support switching to browsing at any time, but the automatic benchmarking mechanisms cannot support this functionality. This makes evaluating the browsing parts of CBIR systems an enormously hard problem. Anyhow, further research on suitable user interfaces for providing more flexible browsing tools are clearly needed. Overall, there has been a notable shift of focus into supporting retrieval by browsing in recent CBIR research.

Automatic image segmentation is an ever fascinating concept for image retrieval and has the potential to lead to prominent steps ahead in the research field. However, since general object recognition is beyond current technologies, short-term interest should be placed mainly in developing methods based on weak segmentation or identifying interest points in the images.

Finally, a crucial task is to develop a common setting for CBIR benchmarking. The CBIR evaluation methodology is still in its infancy and, without a common setting, all published performance evaluations should be taken with a grain of salt. A standardized benchmark will be a significant step ahead as it helps to objectively identify the merits and deficiencies of the current techniques and to guide further research into right directions. Furthermore, a necessary requirement for the development of real image retrieval applications is to test promising approaches with actual test users and retrieval tasks. Due to the inherent human factor in this kind of experiments, a successful evaluation initiative should therefore preferably be an interdisciplinary effort involving computer scientists, IR researchers, and psychologists.

BIBLIOGRAPHY

Addis, M., Boniface, M., Goodall, S., Grimwood, P., Kim, S., Lewis, P., Martinez, K. and Stevenson, A. (2003). Integrated image content and metadata search and retrieval across multiple databases, *Proceedings of International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, USA, pp. 91–100.

Agnew, B., Faloutsos, C., Wang, Z., Welch, D. and Xue, X. (1997). Multi-media indexing over the web, *in* I. K. Sethi and R. J. Jain (eds), *Storage and Retrieval for Image and Video Databases V*, Vol. 3022 of *Proceedings of SPIE*, pp. 72–83.

Aigrain, P., Zhang, H. and Petkovic, D. (1996). Content-based representation and retrieval of visual media: A state-of-the-art review, *Multimedia Tools and Applications* **3**(3): 179–202.

Aksoy, S. and Haralick, R. M. (2001). Feature normalization and likelihood-based similarity measures for image retrieval, *Pattern Recognition Letters* **22**(5): 563–582.

Aksoy, S., Haralick, R. M., Cheikh, F. A. and Gabbouj, M. (2000). A weighted distance approach to relevance feedback, *Proceedings of 15th International Conference on Pattern Recognition (ICPR 2000)*, Vol. 4, Barcelona, Spain, pp. 812–815.

Amsaleg, L. and Gros, P. (2001). Content-based retrieval using local descriptors: Problems and issues from a database perspective, *Pattern Analysis & Applications* **4**(2+3): 108–124.

ANN (2003). University of Washington annotated ground-truth image database, http://www.cs.washington.edu/research/imagedatabase/groundtruth/.

Antani, S., Kasturi, R. and Jain, R. (2002). A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video, *Pattern Recognition* **35**(4): 945–965.

Aslandogan, Y. A. and Yu, C. T. (2000). Experiments in using visual and textual clues for image hunting on the web, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 108–119.

Aslandogan, Y. A., Thier, C., Yu, C. T., Zou, J. and Rishe, N. (1997). Using semantic contents and WordNet in image retrieval, *Proceedings of 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, PA, USA, pp. 286–295.

Assfalg, J., Del Bimbo, A. and Pala, P. (2000). Using multiple examples for content-based image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, Vol. 1, New York City, NY, USA, pp. 335–338.

Bach, J. R., Fuller, C., Gupta, A., Hampapur, A., Horowitz, B., Humphrey, R., Jain, R. and Shu, C.-F. (1996). The Virage image search engine: An open framework for image management, *in* I. K. Sethi and R. J. Jain (eds), *Storage and Retrieval for Image and Video Databases IV*, Vol. 2670 of *Proceedings of SPIE*, pp. 76–87.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*, Addison-Wesley.

Barnard, K. and Forsyth, D. (2001). Learning the semantics of words and pictures, *Proceedings of 8th IEEE International Conference on Computer Vision (ICCV 2001)*, Vancouver, Canada, pp. 408–415.

Barnard, K. and Shirahatti, N. V. (2003). A method for comparing content based image retrieval methods, *Proceedings of SPIE Internet Imaging IV*, Vol. 5018, Santa Clara, CA, USA, pp. 1–8.

Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D., Blei, D. and Jordan, M. I. (2003). Matching words and pictures, *Journal of Machine Learning Research, Special Issue on Machine Learning Methods for Text and Images* **3**: 1107–1135.

Beatty, M. and Manjunath, B. S. (1997). Dimensionality reduction using multi-dimensional scaling for content-based retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 1997)*, Vol. 2, Washington DC, USA, pp. 835–838.

Bellman, R. (1961). *Adaptive control processes: a guided tour*, Princeton University Press.

Benchathlon (2003). The Benchathlon Network WWW site, http://www.benchathlon.net.

Benitez, A. B. and Chang, S.-F. (2002). Semantic knowledge construction from annotated image collections, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, Vol. 2, Lausanne, Switzerland, pp. 205–208.

Benitez, A. B., Beigi, M. and Chang, S.-F. (1998). Using relevance feedback in content-based image metasearch, *IEEE Internet Computing* pp. 59–69.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching, *Communications of the ACM* **18**(9): 509–517.

Berchtold, S., Böhm, C. and Kriegel, H.-P. (1998). The Pyramid-technique: Towards breaking the curse of dimensionality, *Proceedings of ACM SIGMOD International Conference on the Management of Data*, Seattle, WA, USA, pp. 142–153.

Berchtold, S., Keim, D. A. and Kriegel, H.-P. (1996). The X-tree: An indexing structure for high-dimensional data, *Proceedings of 22th International Conference on Very Large Databases (VLDB 96)*, Bombay, India, pp. 28–39.

Berman, A. P. and Shapiro, L. G. (1997). Efficient image retrieval with multiple distance measures, *Storage and Retrieval for Image and Video Databases V*, Vol. 3022 of *Proceedings of SPIE*, San Jose, CA, USA, pp. 12–21.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001). The Semantic Web, *Scientific American* **284**(5): 34–43.

Böhm, C., Berchtold, S. and Keim, D. A. (2001). Searching in high-dimensional spaces—index structures for improving the performance of multimedia databases, *ACM Computing Surveys* **33**(3): 322–373.

Brandt, S. (1999). *Use of shape features in content-based image retrieval*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Brandt, S., Laaksonen, J. and Oja, E. (2002). Statistical shape features for content-based image retrieval, *Journal of Mathematical Imaging and Vision* **17**(2): 187–198.

Breiteneder, C. and Eidenberger, H. (2000). Automatic query generation for content-based image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, Vol. 2, New York City, NY, USA, pp. 705–708.

Bres, S. and Jolion, J.-M. (1999). Detection of interest points for image indexation, *Proceedings of Third International Conference on Visual Information Systems (VISual'99)*, Springer-Verlag, Amsterdam, The Netherlands, pp. 427–434.

Brodatz, P. (1966). *Textures, A photographic album for artists and designers*, Dover Publications.

Brunelli, R. and Mich, O. (2000). Image retrieval by examples, *IEEE Transactions on Multimedia* **2**(3): 164–170.

Brunelli, R. and Mich, O. (2001). Histogram analysis for image retrieval, *Pattern Recognition* **34**(8): 1625–1637.

Caenen, G., Frederix, G., Kuijk, A. A. M., Pauwels, E. J. and Schouten, B. A. M. (2000). Show me what you mean! PARISS: A CBIR-interface that learns by example, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 257–268.

Carson, C., Belongie, S., Greenspan, H. and Malik, J. (2002). Blobworld: Image segmentation using expectation-maximization and its application to image querying, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(8): 1026–1038.

Castelli, V. and Bergman, L. D. (eds) (2002). *Image Databases - Search and Retrieval of Digital Imagery*, John Wiley & Sons, Inc.

Cha, S.-H. and Shihari, S. N. (2002). On measuring the distance between histograms, *Pattern Recognition* **35**(6): 1355–1370.

Chang, N.-S. and Fu, K.-S. (1980). Query-by-Pictorial-Example, *IEEE Transactions on Software Engineering* **6**(6): 519–524.

Chang, S.-F., Chen, W. and Sundaram, H. (1998a). Semantic Visual Templates: Linking visual features to semantics, *Proceedings of IEEE International Conference on Image Processing (ICIP '98)*, Vol. 3, Chicago, IL, USA, pp. 531–535.

Chang, S.-F., Eleftheriadis, A. and McClintock, R. (1998b). Next-generation content representation, creation, and searching for new-media applications in education, *Proceedings of the IEEE* **86**(5): 884–904.

Chang, S.-F., Smith, J. R., Beigi, M. and Benitez, A. (1997). Visual information retrieval from large distributed online repositories, *Communications of the ACM* **40**(12): 63–69.

Chang, S.-K. and Hsu, A. (1992). Image information systems: Where do we go from here?, *IEEE Transactions on Knowledge and Data Engineering* **4**(5): 431–442.

Chen, J.-Y., Bouman, C. A. and Allebach, J. P. (1997). Fast image database search using tree-structured VQ, *Proceedings of IEEE International Conference on Image Processing (ICIP '97)*, Vol. 2, Santa Barbara, CA, USA, pp. 827–830.

Chen, J.-Y., Bouman, C. A. and Dalton, J. C. (2000). Hierarchical browsing and search of large image databases, *IEEE Transactions on Image Processing* **9**(3): 442–455.

Chen, T., Chen, L.-H. and Ma, K.-K. (1999). Colour image indexing using SOM for region-of-interest retrieval, *Pattern Analysis & Applications* **2**: 164–171.

Chen, Y., Zhou, X. S. and Huang, T. S. (2001). One-class SVM for learning in image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 1, Thessaloniki, Greece, pp. 34–37.

Chua, T.-S., Low, W.-C. and Chu, C.-X. (1998). Relevance feedback techniques for color-based image retrieval, *Proceedings of Multimedia Modeling (MMM'98)*, Lausanne, Switzerland, pp. 24–31.

Chuang, G. C.-H. and Kuo, C.-C. J. (1996). Wavelet descriptor of planar curves: Theory and applications, *IEEE Transactions on Image Processing* **5**(1): 56–70.

Ciocca, G. and Schettini, R. (2001). Content-based similarity retrieval of trademarks using relevance feedback, *Pattern Recognition* **34**(8): 1639–1655.

Colombo, C., Del Bimbo, A. and Pala, P. (1999). Semantics in visual information retrieval, *IEEE Multimedia* **6**(3): 38–53.

Comon, P. (1994). Independent component analysis—a new concept?, *Signal Processing* **36**(3): 287–314.

Cox, I. J., Ghosn, J., Miller, M. L., Papathomas, T. V. and Yianilos, P. N. (1997). Hidden annotation in content-based image retrieval, *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Libraries (CBAIVL '97)*, San Juan, Puerto Rico, pp. 76–81.

Cox, I. J., Miller, M. L., Minka, T. P., Papathomas, T. V. and Yianilos, P. N. (2000). The bayesian image retrieval system, PicHunter: Theory, implementation and psychophysical experiments, *IEEE Transactions on Image Processing* **9**(1): 20–37.

Cox, I. J., Miller, M. L., Omohundro, S. M. and Yianilos, P. N. (1996). Target testing and the PicHunter bayesian multimedia retrieval system, *Proceedings of 3rd Forum on Research and Technology Advances in Digital Libraries (ADL'96)*, Washington DC, USA, pp. 66–75.

Csillaghy, A., Hinterberger, H. and Benz, A. O. (2000). Content-based image retrieval in astronomy, *Information Retrieval* **3**(3): 229–241.

Daubechies, I. (1990). The wavelet transform, time-frequency localization and signal analysis, *IEEE Trans. Information Theory* **36**(9): 961–1005.

De Marsicoi, M., Cinque, L. and Levialdi, S. (1999). Indexing pictorial documents by their content: a survey of current techniques, *Image and Vision Computing* **15**(2): 119–141.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K. and Harshman, R. (1990). Indexing by latent semantic analysis, *Journal of the American Society for Information Science* **41**(6): 391–407.

Del Bimbo, A. (1999). *Visual Information Retrieval*, Morgan Kaufmann Publishers, Inc.

Del Bimbo, A. and Pala, P. (1997). Visual image retrieval by elastic matching of user sketches, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**(2): 121–132.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm, *Journal of the Royal Statistical Society* **B-39**(1): 1–38.

Di Lecce, V. and Guerriero, A. (1999). An evaluation of the effectiveness of image features for image retrieval, *Journal of Visual Communication and Image Representation* **10**(4): 351–362.

Doulamis, N. and Doulamis, A. (2001). A recursive optimal relevance feedback scheme for content based image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 2, Thessaloniki, Greece, pp. 741–744.

Duffing, G. and Smaïl, M. (2000). A novel approach for accessing partially indexed image corpora, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 244–256.

Dunckley, L. (2003). *Multimedia Databases: An Object-Relational Approach*, Addison-Wesley.

Eakins, J. P. (2002). Towards intelligent image retrieval, *Pattern Recognition* **35**(1): 3–14.

Eakins, J. P. and Graham, M. E. (1999). Content-based image retrieval. Report to JISC technology applications programme, *Technical report*, Institute for Image Data Research, University of Northumbria at Newcastle. Available at: http://www.unn.ac.uk/iidr/report.html.

Eakins, J. P., Boardman, J. M. and Graham, M. E. (1998). Similarity retrieval of trademark images, *IEEE Multimedia* **5**(2): 53–63.

Egas, R., Huijsmans, N., Lew, M. and Sebe, N. (1999). Adapting k-d trees to visual retrieval, *Proceedings of Third International Conference on Visual Information Systems (VISual'99)*, Springer-Verlag, Amsterdam, The Netherlands, pp. 533–540.

Fellbaum, C. (ed.) (1998). *WordNet: An Electronic Lexical Database*, The MIT Press.

Fidel, R., Hahn, T. B., Rasmussen, E. M. and Smith, P. J. (eds) (1994). *Challenges in Indexing Electronic Text and Images*, ASIS Monograph Series, Learned Information, Inc.

Finkel, R. and Bentley, J. L. (1974). Quad-trees: A data structure for retrieval on composite keys, *ACTA Informatica* **4**(1): 1–9.

Flickner, M., Sawhney, H., Niblack, W. et al. (1995). Query by image and video content: The QBIC system, *IEEE Computer* **28**: 23–31.

Forsyth, D. A., Malik, J., Fleck, M. M. et al. (1996). Finding pictures of objects in large collection of images, *Proceedings of 2nd International Workshop on Object Representation in Computer Vision*, Cambridge, England, pp. 335–360.

Fournier, J. and Cord, M. (2002). Long-term similarity learning in content-based image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2002)*, Vol. 1, Rochester, NY, USA, pp. 441–444.

Fournier, J., Cord, M. and Philipp-Foliguet, S. (2001a). Back-propagation algorithm for relevance feedback in image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 1, Thessaloniki, Greece, pp. 686–689.

Fournier, J., Cord, M. and Philipp-Foliguet, S. (2001b). RETIN: A content-based image indexing and retrieval system, *Pattern Analysis & Applications* **4**(2+3): 153–173.

Frankel, C., Swain, M. J. and Athitsos, V. (1996). Webseer: An image search engine for the world wide web, *Technical Report 96-14*, The University of Chicago.

Freeman, H. (1974). Computer processing of line-drawing images, *Computing Surveys* **6**(1): 57–97.

Garber, S. R. and Grunes, M. B. (1992). The art of search: A study of art directors, *Proceedings of ACM Conference on Human Factors in Computing Systems (ACM CHI 1992)*, Monterey, Canada, pp. 157–163,703.

Geman, D. and Moquet, R. (1999). A stochastic feedback model for image retrieval, *Technical report*, Ecole Polytechnique, France.

Golshani, F. and Park, Y. (1997). Content-based image indexing and retrieval system in ImageRoadMap, *Multimedia Storage and Archiving Systems II*, Vol. 3229 of *Proceedings of SPIE*, Dallas, TX, pp. 194–205.

Gong, Y. (1998). *Intelligent Image Databases: Towards Advanced Image Retrieval*, Kluwer Academic Publishers.

Gudivada, V. N. and Raghavan, V. V. (1995). Content-based image retrieval systems, *IEEE Computer* **28**(9): 18–22.

Gunther, N. J. and Beretta, G. (2000). A benchmark for image retrieval using distributed systems over the internet: BIRDS-I, *Technical Report HPL-2000-162*, HP Labs. Available at: http://www.hpl.hp.com/techreports/2000/HPL-2000-162.html.

Guo, G.-D., Jain, A. K., Ma, W.-Y. and Zhang, H.-J. (2002). Learning similarity measure for natural image retrieval with relevance feedback, *IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing* **13**(4): 811–820.

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

Gupta, A., Santini, S. and Jain, R. (1997). In search of information in visual media, *Communications of the ACM* **40**(12): 35–42.

Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching, *Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data*, Boston, MA, USA, pp. 47–57.

Hafner, J., Sawhney, H. S., Equitz, W., Flickner, M. and Niblack, W. (1995). Efficient color histogram indexing for quadratic form distance functions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **17**(7): 729–736.

Han, J. W. and Guo, L. (2002). A new image retrieval system supporting query by semantics and example, *Proceedings of IEEE International Conference on Image Processing (ICIP 2002)*, Vol. 3, Rochester, NY, USA, pp. 953–956.

Han, K.-A. and Myaeng, S.-H. (1996). Image organization and retrieval with automatically constructed feature vectors, *Proceedings of 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Zurich, Switzerland, pp. 157–165.

Haralick, R., Shanmugam, K. and Dinstein, I. (1973). Textual features for image classification, *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-3**(6): 610–621.

Harman, D. (1992). Overview of the first Text REtrieval Conference (TREC-1), *NIST Special Publication 500-207: The First Text REtrieval Conference (TREC-1)*, Gaithersburg, MD, USA, pp. 1–20.

Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation*, second edn, Prentice Hall, Inc.

Heisterkamp, D. R. (2002). Building a latent semantic index of an image database from patterns of relevance feedback, *Proceedings of the 16th International Conference on Pattern Recognition (ICPR 2002)*, Vol. 4, Quebec, Canada, pp. 134–137.

Hiroike, A., Musha, Y., Sugimoto, A. and Mori, Y. (1999). Visualization of information spaces to retrieve and browse image data, *Proceedings of Third International Conference on Visual Information Systems (VISual'99)*, Springer-Verlag, Amsterdam, The Netherlands, pp. 155–162.

Hong, P., Tian, Q. and Huang, T. S. (2000). Incorporate Support Vector Machines to content-based image retrieval with relevance feedback, *Proceedings of IEEE International Conference on Image Processing (ICIP 2000)*, Vol. 3, Vancouver, Canada, pp. 750–753.

Honkela, T., Kaski, S., Lagus, K. and Kohonen, T. (1997). WEBSOM—self-organizing maps of document collections, *Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6*, Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland, pp. 310–315.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components, *Journal of Educational Psychology* **24**: 498–520.

Hsieh, I.-S. and Fan, K. C. (2001). Multiple classifiers for color flag and trademark image retrieval, *IEEE Transactions on Image Processing* **10**(6): 938–950.

Hsu, W., Chua, T. S. and Pung, H. K. (1995). An integrated color-spatial approach to content-based image retrieval, *Proceedings of 3rd International ACM Multimedia Conference (ACM MM '95)*, San Francisco, CA, USA, pp. 305–313.

Hu, M.-K. (1962). Visual pattern recognition by moment invariants, *IRE Transactions on Information Theory* **8**: 179–187.

Huang, J., Kumar, S. R. and Mitra, M. (1997a). Combining supervised learning with color correlograms for content-based image retrieval, *Proceedings of 5th International ACM Multimedia Conference (ACM MM '97)*, Seattle, WA, USA, pp. 325–334.

Huang, J., Kumar, S. R., Mitra, M., Zhu, W.-J. and Zabih, R. (1997b). Image indexing using color correlograms, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97)*, San Juan, Puerto Rico, pp. 762–768.

Huang, T. S. and Zhou, X. S. (2001). Image retrieval with relevance feedback: From heuristic weight adjustment to optimal learning methods, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 3, Thessaloniki, Greece, pp. 2–5.

Huang, T. S., Mehrotra, S. and Ramchandran, K. (1996). Multimedia analysis and retrieval system (MARS) project, *Proceedings of 33rd Annual Clinic on Library Application on Data Processing - Digital Image Access and Retrieval*, Urbana-Champaign, IL, USA.

Hussain, M., Eakins, J. and Sexton, G. (2002). Visual clustering of trademarks using the self-organizing map, *Proceedings of The Challenge of Image and Video Retrieval (CIVR 2002)*, London, UK, pp. 147–156.

Hyvärinen, A., Karhunen, J. and Oja, E. (2001). *Independent Component Analysis*, John Wiley & Sons.

Idris, F. and Panchanathan, S. (1997). Review of image and video indexing techniques, *Journal of Visual Communication and Image Representation* **8**(2): 146–166.

Iivarinen, J. and Pakkanen, J. (2002). Content-based retrieval of defect images, *Proceedings of Advanced Concepts for Intelligent Vision Systems (ACIVS 2002)*, Ghent, Belgium, pp. 62–67.

Ingwersen, P. (1992). *Information Retrieval Interaction*, Taylor Graham Publishing. Available at: http://www.db.dk/pi/iri.

Ishikawa, Y., Subramanya, R. and Faloutsos, C. (1998). MindReader: Querying databases through multiple examples, *Proceedings of 24rd International Conference on Very Large Data Bases (VLDB'98)*, New York City, NY, USA, pp. 218–226.

Iyengar, G. and Lippman, A. (1998). Clustering images using relative entropy for efficient retrieval, *Proceedings of International Workshop on Very Low Bitrate Video Coding (VLBV'98)*, Urbana, IL, USA.

Jain, A. K. (1989). *Fundamentals of Digital Image Processing*, Prentice Hall, Inc.

Jain, A. K. and Vailaya, A. (1998). Shape-based retrieval: A case study with trademark image databases, *Pattern Recognition* **31**(9): 1369–1390.

Jain, R. (1997). Content-centric computing in visual systems, *Proceedings of 9th International Conference on Image Analysis and Processing (ICIAP'97)*, Vol. II, Florence, Italy, pp. 1–13.

Jeong, S., Kim, K., Chun, B., Jaeyeon and Bae, Y. J. (1999). An effective method for combining multiple features of image retrieval, *Proceedings of IEEE Region 10 Conference (TENCON'99)*, Vol. 2, Cheju, Korea, pp. 982–985.

Johansson, B. (2000). A survey on: Contents based search in image databases, *Technical report*, Department of Electrical Engineering, Linköping University. Available at: http://www.isy.liu.se/cvl/Projects/VISIT-bjojo/.

Jörgensen, C. and Jörgensen, P. (2002). Testing a vocabulary for image indexing and ground truthing, *Internet Imaging III*, Vol. 4672 of *Proceedings of SPIE*, San Jose, CA, USA, pp. 207–215.

Jose, J. M., Furner, J. and Harper, D. J. (1998). Spatial querying for image retrieval: a user-oriented evaluation, *Proceedings of 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Melbourne, Australia, pp. 232–240.

Kaski, S. (1998). Dimensionality reduction by random mapping: Fast similarity method for clustering, *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN98)*, Vol. 1, Anchorage, AK, USA, pp. 413–418.

Kaski, S., Kangas, J. and Kohonen, T. (1998). Bibliography of self-organizing map (SOM) papers: 1981-1997, *Neural Computing Surveys* **1**(3&4): 1–176.

Kato, T. (1992). Database architecture for content-based image retrieval, *in* I. K. Sethi and R. J. Jain (eds), *Image Storage and Retrieval Systems*, Vol. 1662 of *Proceedings of SPIE*, San Jose, CA, USA, pp. 112–123.

Khotanzad, A. and Hong, Y. H. (1990). Invariant image recognition by zernike moments, *IEEE Transaction on Pattern Analysis and Machine Intelligence* **12**(5): 489–497.

King, I. and Jin, Z. (2001). Relevance feedback content-based image retrieval using query distribution estimation based on maximum entropy principle, *Proceedings of the 8th International Conference on Neural Information Processing (ICONIP 2001)*, Vol. 2, Shanghai, China, pp. 699–704.

King, I. and Lau, T. K. (1997). Competitive learning clustering for information retrieval in image databases, *Proceedings of International Conference on Neural Information Processing and Intelligent Information Systems (ICONIP 1997)*, Dunedin, New Zealand, pp. 906–909.

Kohonen, T. (1982). Self-organizing formation of topologically correct feature maps, *Biological Cybernetics* **43**(1): 59–69.

Kohonen, T. (2001). *Self-Organizing Maps*, Vol. 30 of *Springer Series in Information Sciences*, third edn, Springer-Verlag.

Kohonen, T., Kaski, S., Lagus, K., Salojärvi, J., Honkela, J., Paatero, V. and Saarela, A. (2000). Self organization of a massive text document collection, *IEEE Transactions on Neural Networks* **11**(3): 574–585.

Kohonen, T., Oja, E., Simula, O., Visa, A. and Kangas, J. (1996). Engineering applications of the self-organizing map, *Proceedings of the IEEE* **84**(10): 1358–84.

Koikkalainen, P. (1994). Progress with the tree-structured self-organizing map, *in* A. G. Cohn (ed.), *11th European Conference on Artificial Intelligence*, European Committee for Artificial Intelligence (ECCAI), John Wiley & Sons, Ltd.

Koikkalainen, P. and Oja, E. (1990). Self-organizing hierarchical feature maps, *Proceedings of International Joint Conference on Neural Networks*, Vol. II, San Diego, CA, USA, pp. 279–284.

Kolenda, T., Hansen, L. K., Larsen, J. and Winther, O. (2002). Independent component analysis for understanding multimedia content, *Proceedings of Neural Networks for Signal Processing XII*, Martigny, Switzerland, pp. 757–766.

Koskela, M. (1999). *Content-based image retrieval with self-organizing maps*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Kraaijveld, M. A., Mao, J. and Jain, A. K. (1992). A non-linear projection method based on Kohonen's topology preserving maps, *Proceedings of the 11th International Conference on Pattern Recognition (ICPR 1992), Conference B: Pattern Recognition Methodology and Systems*, Vol. 2, Orono, ME, USA, pp. 41–45.

Kriegel, N. B. H.-P., Scheider, R. and Seeger, B. (1990). The R*-tree: an efficient and robust access method for points and rectangles, *Proceedings of ACM SIGMOD International Conference on the Management of Data*, Atlantic City, NJ, USA, pp. 322–331.

Krishnamachari, S. and Abdel-Mottaleb, M. (1999). Image browsing using hierarchical clustering, *Proceedings of 4th IEEE International Symposium on Computers and Communications*, Red Sea, Egypt, pp. 301–307.

Kruskal, J. B. (1964). Multidimensional scaling, *Psychometrika* **29**(1): 1–27.

Kulkarni, S., Srinivasan, B. and Ramakrishna, M. V. (1999). Vector-space image model (VSIM) for content-based retrieval, *Proceedings of 10th International Workshop on Database and Expert Systems Applications (DEXA'99)*, Florence, Italy, pp. 899–903.

La Cascia, M., Sethi, S. and Sclaroff, S. (1998). Combining textual and visual cues for content-based image retrieval on the world wide web, *Proceedings of IEEE International Workshop on Content-based Access of Image and Video Libraries (CBAIVL '98)*, Santa Barbara, CA, USA, pp. 24–28.

Laakso, S. (2000). *Implementation of content-based www image search engine*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Laakso, S., Laaksonen, J., Koskela, M. and Oja, E. (2001). Self-organizing maps of web link information, *in* N. Allinson, H. Yin, L. Allinson and J. Slack (eds), *Advances in Self-Organising Maps*, Springer, Lincoln, England, pp. 146–151.

Laaksonen, J., Koskela, M. and Oja, E. (2003). Probability interpretation of distributions on SOM surfaces, *Proceedings of Workshop on Self-Organizing Maps (WSOM'03)*, Hibikino, Kitakyushu, Japan.

Lagus, K. (2002). Text retrieval using self-organized document maps, *Neural Processing Letters* **15**(1): 21–29.

Le Saux, B. and Boujemaa, N. (2002). Unsupervised categorization for image database overview, *Proceedings of 5th International Conference on Visual Information System*, HsinChu, Taiwan, pp. 163–174.

Leung, C. H. C. and Ip, H. H. S. (2000). Benchmarking for visual information search, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 442–456.

Lew, M. S. (2000). Next-generation web searches for visual content, *IEEE Computer* **33**(11): 46–53.

Lew, M. S. and Denteneer, D. (2001). Fisher keys for content based retrieval, *Image and Vision Computing* **19**(8): 561–566.

Lew, M. S. (ed.) (2001). *Principles of Visual Information Retrieval*, Springer-Verlag.

Li, B., Chang, E. and Li, C.-S. (2001). Learning image query concepts via intelligent sampling, *Proceedings of International Conference on Multimedia and Exposition*, Tokyo, Japan.

Li, B., Chang, E. and Wu, C.-T. (2002a). DPF – a perceptual distance function for image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2002)*, Vol. 2, Rochester, NY, USA, pp. 597–600.

Li, M., Chen, Z. and Zhang, H.-J. (2002b). Statistical correlation analysis in image retrieval, *Pattern Recognition* **35**(12): 2687–2693.

Linde, Y., Buzo, A. and Gray, R. M. (1980). An algorithm for vector quantizer design, *IEEE Transactions on Communications* **28**(1): 84–95.

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

Loupias, E. and Sebe, N. (2000). Wavelet-based salient points: Applications to image retrieval using color and texture features, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 223–232.

Lu, G. (2002). Techniques and data structures for efficient multimedia retrieval based on similarity, *IEEE Transactions on Multimedia* **4**(3): 372–384.

Lu, G. and Teng, S. (1999). A novel image retrieval technique based on vector quantization, *Proceedings of International Conference on Computational Intelligence for Modelling, Control, and Automation*, Vienna, Austria, pp. 36–41.

Ma, W. Y. and Manjunath, B. S. (1996). Texture features and learning similarity, *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'96)*, San Francisco, CA, USA, pp. 425–430.

Ma, W. Y. and Manjunath, B. S. (1997). NETRA: A toolbox for navigating large image databases, *Proceedings of IEEE International Conference on Image Processing (ICIP '97)*, Vol. 1, Santa Barbara, CA, USA, pp. 568–571.

Ma, W.-Y. and Zhang, H. (1998). Benchmarking of image features for content-based retrieval, *Proceedings of the 32nd Asilomar Conference on Signals, Systems & Computers*, Vol. 1, Pacific Grove, Canada, pp. 253–257.

MacArthur, S. D., Brodley, C. E. and Shyu, C.-R. (2000). Relevance feedback decision trees in content-based image retrieval, *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL 2000)*, Hilton Head Island, SC, USA, pp. 68–72.

Manjunath, B. S., Salembier, P. and Sikora, T. (eds) (2002). *Introduction to MPEG-7: Multimedia Content Description Interface*, John Wiley & Sons Ltd.

Mao, J. and Jain, A. K. (1992). Texture classification and segmentation using multiresolution simultaneous autoregressive models, *Pattern Recognition* **25**(2): 173–188.

Markkula, M. and Sormunen, E. (2000). End-user searching challenges indexing practices in the digital newspaper photo archive, *Information Retrieval* **1**(4): 259–285.

Matinmikko, E. (2002). *Kuvatietokannan selailujärjestelmä (in Finnish)*, Master's thesis, Department of Electrical Engineering, University of Oulu, Finland.

Meilhac, C. and Nastar, C. (1999). Relevance feedback and category search in image databases, *Proceedings of IEEE International Conference on Multimedia Computing and Systems (ICMCS'99)*, Vol. 1, Florence, Italy, pp. 512–517.

Minka, T. P. and Picard, R. W. (1997). Interactive learning using a 'society of models', *Pattern Recognition* **30**(4): 565–581.

MPEG (2001). MPEG-7 visual part of the eXperimentation Model (version 9.0). ISO/IEC JTC1/SC29/WG11 N3914.

MPEG (2002). MPEG-7 Overview (version 8.0). ISO/IEC JTC1/SC29/WG11 N4980.

MPEG (2003). MPEG-7 eXperimentation Model (XM) primary source repository, http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html.

MRML (2003). Multimedia Retrieval Markup Language (MRML) WWW site, http://www.mrml.net.

Müller, H., Marchand-Maillet, S. and Pun, T. (2002). The truth about Corel – evaluation in image retrieval, *Proceedings of The Challenge of Image and Video Retrieval (CIVR 2002)*, London, UK, pp. 38–49.

Müller, H., Müller, W., Marchand-Maillet, S. and Pun, T. (2000a). Strategies for positive and negative relevance feedback in image retrieval, *Proceedings of 15th International Conference on Pattern Recognition (ICPR 2000)*, Vol. 1, Barcelona, Spain, pp. 1043–1046.

Müller, H., Müller, W., Squire, D. M., Marchand-Maillet, S. and Pun, T. (2000b). Long-term learning from user behavior in content-based image retrieval, *Technical Report 00.04*, Computer Vision Group, University of Geneva, Geneva, Switzerland.

Müller, H., Müller, W., Squire, D. M., Marchand-Maillet, S. and Pun, T. (2001). Performance evaluation in content-based image retrieval: overview and proposals, *Pattern Recognition Letters* **22**(5): 593–601.

Müller, W., Müller, H., Marchand-Maillet, S., Pun, T., Squire, D. M., Pečenović, Z., Giess, C. and de Vries, A. P. (2000c). MRML: A communication protocol for content-based image retrieval, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 300–311.

Müller, W., Squire, D. M., Müller, H. and Pun, T. (1999). Hunting moving targets: an extension to Bayesian methods in multimedia databases, *Proceedings of Multimedia Storage and Archiving Systems IV (VV02)*, Vol. 3846 of *SPIE Proceedings*, Boston, MA, USA.

Najjar, M., Ambroise, C. and Cocquerez, J. P. (2003). Image retrieval using mixture models and EM algorithm, *Proceedings of 13th Scandinavian Conference on Image Analysis (SCIA 2003)*, Halmstad, Sweden, pp. 1114–1121.

Nakazato, M., Manola, L. and Huang, T. S. (2002). ImageGrouper: Search, annotate and organize images by groups, *Proceedings of 5th International Conference on Visual Information System*, HsinChu, Taiwan, pp. 129–142.

Naphade, M. R. and Huang, T. S. (2002). Extracting semantics from audiovisual content: The final frontier in multimedia retrieval, *IEEE Transactions on Neural Networks* **13**(4): 793–810.

Naphade, M. R., Kristjansson, T., Frey, B. and Huang, T. S. (1998). Probabilistic multimedia objects (Multijects): A novel approach to video indexing and retrieval in multimedia systems, *Proceedings of IEEE International Conference on Image Processing (ICIP '98)*, Vol. 3, Chicago, IL, USA, pp. 536–540.

Nastar, C., Mitschke, M. and Meilhac, C. (1998). Efficient query refinement for image retrieval, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, CA, USA, pp. 547–552.

Natsev, A. and Smith, J. R. (2002). A study of image retrieval by anchoring, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, Vol. 2, Lausanne, Switzerland, pp. 421–424.

Nene, S. A., Nayar, S. K. and Murase, H. (1996). Columbia Object Image Library (COIL-100), *Technical Report CUCS-006-96*, Columbia University, Department of Computer Science.

Neumann, J., Samet, H. and Soffer, A. (2002). Integration of local and global shape analysis for logo classification, *Pattern Recognition Letters* **23**(12): 1449–1457.

Newsam, S., Sumengen, B. and Manjunath, B. S. (2001). Category-based image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 3, Thessaloniki, Greece, pp. 596–599.

Niblack, W., Barber, R., Equitz, W., Flickner, M., Petkovic, E. H. G. D., Yanker, P., Faloutsos, C. and Taubin, G. (1993). The QBIC project: Querying images by content using color, texture and shape, *Storage and Retrieval for Image and Video Databases (SPIE)*, Vol. 1908 of *SPIE Proceedings Series*, San Jose, CA, USA.

Nielsen, J. (1994). *Usability Engineering*, Morgan Kaufmann Publishers, Inc.

Ogle, V. E. and Stonebraker, M. (1995). Chabot: Retrieval from a relational database of images, *IEEE Computer* **28**: 40–48.

Oh, K. S., Zaher, A. and Kim, P. K. (2002). Fast k-NN image search with self-organizing maps, *Proceedings of The Challenge of Image and Video Retrieval (CIVR 2002)*, London, UK, pp. 299–308.

Oja, E., Laaksonen, J., Koskela, M. and Brandt, S. (1999). Self-organizing maps for content-based image retrieval, *in* E. Oja and S. Kaski (eds), *Kohonen Maps*, Elsevier, pp. 349–362.

Oja, M., Kaski, S. and Kohonen, T. (2003). Bibliography of Self-Organizing Map (SOM) papers: 1998-2001 addendum, *Neural Computing Surveys* **3**(1): 1–156.

Ojala, T., Aittola, M. and Matinmikko, E. (2002). Empirical evaluation of MPEG-7 XM color descriptors in content-based retrieval of semantic image categories, *Proceedings of the 16th International Conference on Pattern Recognition (ICPR 2002)*, Vol. 2, Quebec, Canada, pp. 1021–1024.

Ojala, T., Pietikäinen, M. and Harwood, D. (1996). A comparative study of texture measures with classification based on feature distributions, *Pattern Recognition* **29**(1): 51–59.

Ong, S. H., Yeo, N. C., Lee, K. H., Venkatesh, Y. V. and Cao, D. M. (2002). Segmentation of color images using a two-stage self-organizing network, *Image and Vision Computing* **20**(4): 279–289.

Ornager, S. (1997). Image retrieval: Theoretical analysis and empirical user studies on accessing information in images, *Proceedings of American Society for Information Science and Technology Annual Meeting (ASIS'97)*, Washington DC, USA, pp. 202–211.

Ortega-Binderberger, M. and Mehrotha, S. (2003). Relevance feedback in multimedia databases, *in* B. Furht and O. Marquez (eds), *Handbook of Video Databases: Design and Applications*, CRC Press, chapter 23.

Pakkanen, J. (2002). *Sisältöpohjainen haku paperivirhetietokannassa PicSOM-järjestelmän avulla (in Finnish)*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Pass, G., Zabih, R. and Miller, J. (1996). Comparing images using color coherence vectors, *Proceedings of the 4th International ACM Multimedia Conference (ACM MM '96)*, Boston, MA, USA, pp. 65–73.

Pastra, K., Saggion, H. and Wilks, Y. (2003). Intelligent indexing of crime scene photographs, *IEEE Intelligent Systems* **18**(1): 55–61.

Pečenović, Z., Do, M. N., Vetterli, M. and Pu, P. (2000). Integrated browsing and searching of large image collections, *Proceedings of Fourth International Conference on Visual Information Systems (VISual 2000)*, Lyon, France, pp. 279–289.

Pentland, A., Picard, R. W. and Sclaroff, S. (1994). Photobook: Tools for content-based manipulation of image databases, *Storage and Retrieval for Image and Video Databases II*, Vol. 2185 of *Proceedings of SPIE*, San Jose, CA, USA, pp. 34–47.

Persoon, E. and Fu, K. S. (1977). Shape discrimination using Fourier descriptors, *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-7**(3): 170–179.

Pfund, T. and Marchand-Maillet, S. (2002). A dynamic multimedia annotation tool, *Internet Imaging III*, Vol. 4672 of *Proceedings of SPIE*, San Jose, CA, USA, pp. 216–224.

Picard, R. W., Minka, T. P. and Szummer, M. (1996). Modeling user subjectivity in image libraries, *Technical Report #382*, M.I.T Media Laboratory.

Qian, F., Li, M., Zhang, L., Zhang, H.-J. and Zhang, B. (2002). Gaussian mixture model for relevance feedback in image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, Vol. 1, Lausanne, Switzerland, pp. 229–232.

Qian, F., Zhang, B. and Lin, F. (2003). Constructive learning algorithm-based RBF network for relevance feedback in image retrieval, *Proceedings of International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, USA, pp. 352–361.

Qiu, G. (2002). Indexing chromatic and achromatic patterns for content-based colour image retrieval, *Pattern Recognition* **35**(8): 1675–1686.

Rao, A., Srihari, R. K., Zhu, L. and Zhang, A. (2002). A method for measuring the complexity of image databases, *IEEE Transactions on Multimedia* **4**(2): 160–173.

Ravi Kanth, K. V., Agrawal, D., El Abbadi, A. and Singh, A. (1999). Dimensionality reduction for similarity searching in dynamic databases, *Computer Vision and Image Understanding, Special Issue on Content-Based Access for Image and Video Libraries* **75**(1/2): 59–72.

Ren, C. and Means, R. W. (1998). Context vector approach to image retrieval, *Applications of Artificial Neural Networks in Image Processing III*, Vol. 3307 of *Proceedings of SPIE*, pp. 137–141.

Rocchio, J. J. (1971). Relevance feedback in information retrieval, *in* G. Salton (ed.), *The SMART retrieval system: Experiments in automatic document processing*, Prentice-Hall, pp. 313–323.

Rodden, K. and Wood, K. (2003). How do people manage their digital photographs?, *Proceedings of ACM Conference on Human Factors in Computing Systems (ACM CHI 2003)*, Fort Lauderdale, FL, USA, pp. 409–416.

Rodden, K., Basalaj, W., Sinclair, D. and Wood, K. (1999). Evaluating a visualisation of image similarity as a tool for image browsing, *Proceedings of IEEE Symposium on Information Visualisation (Info Vis'99)*, San Francisco, CA, USA, pp. 36–43,143.

Rodden, K., Basalaj, W., Sinclair, D. and Wood, K. (2001). Does organisation by similarity assist image browsing?, *Proceedings of ACM Conference on Human Factors in Computing Systems (ACM CHI 2001)*, Seattle, WA, USA, pp. 190–197.

Rubner, Y. (1999). *Perceptual metrics for image database navigation*, PhD thesis, Stanford University.

Rubner, Y., Puzicha, J., Tomasi, C. and Buhmann, J. M. (2001). Empirical evaluation of dissimilarity measures for color and texture, *Computer Vision and Image Understanding* **84**(1): 25–43.

Rubner, Y., Tomasi, C. and Guibas, L. J. (1998). The Earth Mover's Distance as a metric for image retrieval, *Technical Report CS-TN-98-86*, Stanford University.

Rui, Y., Huang, T. S., , M. O. and Mehrotra, S. (1998). Relevance feedback: A power tool in interactive content-based image retrieval, *IEEE Transactions on Circuits and Systems for Video Technology* **8**(5): 644–655.

Rui, Y., Huang, T. S. and Chang, S.-F. (1999). Image retrieval: Current techniques, promising directions, and open issues, *Journal of Visual Communication and Image Representation* **10**(1): 39–62.

Rui, Y., Huang, T. S. and Mehrotra, S. (1997). Content-based image retrieval with relevance feedback in MARS, *Proceedings of IEEE International Conference on Image Processing (ICIP '97)*, Santa Barbara, CA, USA, pp. 815–818.

Rummukainen, M. (2003). *Implementing multimedia retrieval markup language for image retrieval systems' comparison*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Rummukainen, M., Laaksonen, J. and Koskela, M. (2003). An efficiency comparison of two content-based image retrieval systems, GIFT and PicSOM, *Proceedings of International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, USA, pp. 500–509.

Salton, G. (1992). The state of retrieval system evaluation, *Information Processing & Management* **28**(4): 441–449.

Salton, G. and McGill, M. J. (1983). *Introduction to Modern Information Retrieval*, Computer Science Series, McGraw-Hill.

Salton, G. (ed.) (1971). *The SMART retrieval system: Experiments in automatic document processing*, Prentice-Hall.

Samet, H. and Soffer, A. (1996). MARCO: MAp Retrieval by COntent, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **18**(8): 783–798.

Santini, S. (2001). *Exploratory Image Databases*, Academic Press.

Santini, S. and Jain, R. (1999). Similarity measures, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21**(9): 1–13.

Santini, S. and Jain, R. (2000). Integrated browsing and querying for image databases, *IEEE Multimedia* **7**(3): 26–39.

Santini, S., Gupta, A. and Jain, R. (2001). Emergent semantics through interaction in image databases, *IEEE Transactions on Knowledge and Data Engineering* **13**(3): 337–351.

Schettini, R., Ciocca, G. and Gagliardi, I. (1999). Content-based color image retrieval with relevance feedback, *Proceedings of IEEE International Conference on Image Processing (ICIP '99)*, Vol. 3, Kobe, Japan, pp. 75–79.

Schmid, C. and Mohr, R. (1997). Local grayvalue invariants for image retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**(5): 530–535.

Schreiber, A. T., Dubbelham, B., Wielemaker, J. and Wielinga, B. (2001). Ontology-based photo annotation, *IEEE Intelligent Systems* **16**(3): 66–74.

Sclaroff, S., La Cascia, M., Sethi, S. and Taycher, L. (1999). Unifying textual and visual cues for content-based image retrieval on the World Wide Web, *Computer Vision and Image Understanding, Special Issue on Content-Based Access for Image and Video Libraries* **75**(1/2): 86–98.

Sethi, I. K. and Coman, I. (1999). Image retrieval using hierarchical self-organizing feature maps, *Pattern Recognition Letters* **20**(11): 1337–1345.

Sheikholeslami, G., Chang, W. and Zhang, A. (1998). Semantic clustering and querying on heterogeneous features for visual data, *Proceedings of 6th International ACM Multimedia Conference (ACM MM'98)*, Bristol, UK, pp. 3–12.

Shyu, C.-R., Brodley, C., Kak, A., Kosaka, A., Aisen, A. M. and Broderick, L. S. (1999). ASSERT: A physician-in-the-loop content-based retrieval system for HRCT image databases, *Computer Vision and Image Understanding* **75**(1/2): 111–132.

Sjöberg, M., Laaksonen, J. and Viitaniemi, V. (2003). Using image segments in PicSOM CBIR system, *Proceedings of 13th Scandinavian Conference on Image Analysis (SCIA 2003)*, Halmstad, Sweden, pp. 1106–1113.

Smeaton, A. F. and Over, P. (2003). TRECVID: Benchmarking the effectiveness of information retrieval tasks on digital video, *Proceedings of International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, USA, pp. 19–27.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A. and Jain, R. (2000). Content-based image retrieval at the end of the early years, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(12): 1349–1380.

Smith, J. R. (1997). *Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression*, PhD thesis, Graduate School of Arts and Sciences, Columbia University.

Smith, J. R. (1998). Image retrieval evaluation, *Proceeding of IEEE International Workshop on Content-Based Access of Image and Video Libraries (CBAIVL '98)*, Santa Barbara, CA, USA, pp. 112–113.

Smith, J. R. and Chang, S.-F. (1995). Single color extraction and image query, *Proceedings of IEEE International Conference on Image Processing (ICIP '95)*, Vol. 3, Washington DC, USA, pp. 528–531.

Smith, J. R. and Chang, S.-F. (1996). VisualSEEk: A fully automated content-based image query system, *Proceedings of the 4th International ACM Multimedia Conference (ACM MM '96)*, Boston, MA, USA, pp. 87–98.

Smith, T. R. (1996). A digital library for geographically referenced materials, *IEEE Computer* **29**(5): 54–60.

Song, B. C., Kim, M. J. and Ra, J. B. (2001). A fast multiresolution feature matching algorithm for exhaustive search in large image databases, *IEEE Transactions on Circuits and Systems for Video Technology* **11**(5): 673–678.

Squire, D. M. and Pun, T. (1998). Assessing agreement between human and machine clusterings of image databases, *Pattern Recognition* **31**(12): 1905–1919.

Squire, D., Müller, H. and Müller, W. (1999a). Improving response time by search pruning in a content-based image retrieval system, using inverted file techniques, *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Libraries (CBAIVL '99)*, Fort Collins, CO, USA, pp. 45–49.

Squire, D., Müller, W. and Müller, H. (1999b). Relevance feedback and term weighting schemes for content-based image retrieval, *Proceedings of Third International Conference on Visual Information Systems (VISual'99)*, Springer-Verlag, Amsterdam, The Netherlands, pp. 549–556.

Squire, D., Müller, W., Müller, H. and Pun, T. (2000). Content-based query of image databases: inspirations from text retrieval, *Pattern Recognition Letters* **21**(13-14): 1193–1198.

Srihari, R. K. and Zhang, Z. (2000). Show&Tell: A semi-automated image annotation system, *IEEE Multimedia* **7**(3): 61–71.

Stan, D. and Sethi, I. K. (2003). *e*ID: a system for exploration of image databases, *Information Processing & Management* **39**(3): 335–361.

Stricker, M. and Dimai, A. (1996). Color indexing with weak spatial constraints, *Storage and Retrieval for Image and Video Databases IV (SPIE)*, Vol. 2670 of *SPIE Proceedings Series*, San Diego, CA, USA, pp. 29–40.

Stricker, M. and Orengo, M. (1995). Similarity of color images, *Storage and Retrieval for Image and Video Databases III (SPIE)*, Vol. 2420 of *SPIE Proceedings Series*, San Jose, CA, USA, pp. 381–392.

Su, Z., Li, S. and Zhang, H. (2001). Extraction of feature subspaces for content-based retrieval using relevance feedback, *Proceedings of 9th ACM International Conference on Multimedia (ACM MM '01)*, Ottawa, Canada, pp. 98–106.

Suganthan, P. N. (2002). Shape indexing using self-organizing maps, *IEEE Transactions on Neural Networks* **13**(4): 835–840.

Sull, S., Oh, J., Oh, S., Song, S. M.-H. and Lee, S. W. (2000). Relevance graph-based image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, Vol. 2, New York City, NY, USA, pp. 713–716.

Swain, M. J. and Ballard, D. H. (1991). Color indexing, *International Journal of Computer Vision* **7**(1): 11–32.

Sychay, G., Chang, E. and Goh, K. (2002). Effective image annotation via active learning, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, Vol. 1, Lausanne, Switzerland, pp. 209–212.

Szummer, M. and Picard, R. W. (1998). Indoor-outdoor image classification, *Proceedings of IEEE International Workshop on Content-Based Access of Image and Video Database*, Bombay, India, pp. 42–51.

Tamura, H. and Yokoya, N. (1984). Image database systems: A survey, *Pattern Recognition* **17**(1): 29–43.

Tamura, H., Mori, S. and Yamawaki, T. (1978). Texture features corresponding to visual perception, *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-8**(6): 460–473.

Taycher, L., La Cascia, M. and Sclaroff, S. (1997). Image digestion and relevance feedback in the ImageRover WWW search engine, *Proceedings of International Conference on Visual Information Systems (VISual'97)*, San Diego, CA, USA, pp. 85–92.

TC12 (2003). IAPR Technical Committee TC12 benchmark image database, http://sci.vu.edu.au/˜clement/tc-12/benchmark.htm.

Tian, Q., Moghaddam, B. and Huang, T. S. (2002). Visualization, estimation and user-modeling for interactive browsing of image libraries, *Proceedings of The Challenge of Image and Video Retrieval (CIVR 2002)*, London, UK, pp. 7–16.

Tieu, K. and Viola, P. (2000). Boosting image retrieval, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2000)*, Vol. 1, Hilton Head Island, SC, USA, pp. 228–235.

Tong, S. and Chang, E. (2001). Support Vector Machine active learning for image retrieval, *Proceedings of the 9th International ACM Multimedia Conference (ACM MM '01)*, Ottawa, Canada, pp. 107–118.

TREC (2003). TREC NIST WWW site, http://trec.nist.gov.

TRECVID (2003). TREC Video Retrieval Evaluation (TRECVID) WWW home page, http://www-nlpir.nist.gov/projects/trecvid/.

Turner, M. R. (1986). Texture discrimination by Gabor functions, *Biological Cybernetics* **55**(2/3): 71–82.

Ultsch, A. and Siemon, H. P. (1990). Kohonen's self organizing feature maps for exploratory data analysis, *Proceedings of International Neural Network Conference (INNC-90)*, Paris, France, pp. 305–308.

Vailaya, A., Figueiredo, M. A. T., Jain, A. K. and Zhang, H.-J. (2001). Image classification for content-based indexing, *IEEE Transactions on Image Processing* **10**(1): 117–130.

Vailaya, A., Jain, A. and Zhang, H. J. (1998). On image classification: City images vs. landscapes, *Pattern Recognition* **31**(12): 1921–1935.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*, Springer-Verlag.

Vasconcelos, N. and Kunt, M. (2001). Content-based retrieval from image databases: Current solutions and future directions, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 3, Thessaloniki, Greece, pp. 6–9.

Vasconcelos, N. and Lippman, A. (1998). Embedded mixture modeling for efficient probabilistic content-based indexing and retrieval, *Multimedia Storage and Archiving Systems III*, Vol. 3527 of *Proceedings of SPIE*, Boston, MA, USA.

Vasconcelos, N. and Lippman, A. (1999). Learning from user feedback in image retrieval systems, *Advances in Neural Information Processing Systems 12: Proceedings of the 1999 Conference (NIPS*99)*, Denver, CO, USA, pp. 977–983.

Vasconcelos, N. and Lippman, A. (2000). Learning over multiple temporal scales in image databases, *Proceedings of Sixth European Conference on Computer Vision (ECCV'2000)*, Vol. 1, Dublin, Ireland, pp. 33–47.

Veltkamp, R. C. and Tanase, M. (2000). Content-based image retrieval systems: A survey, *Technical Report 2000-34*, Utrecht University, Information and Computing Sciences, Utrecht, The Netherlands. Available at: http://www.cs.uu.nl/research/techreps/UU-CS-2000-34.html.

Venters, C. C. and Cooper, M. (2000). A review of content-based image retrieval systems, *Technical report*, University of Manchester. Available at: http://www.jtap.ac.uk/reports/.

Vesanto, J. and Alhoniemi, E. (2000). Clustering of the self-organizing map, *IEEE Transactions on Neural Networks* **11**(3): 586–600.

Viitaniemi, V. (2002). *Image segmentation in content-based image retrieval*, Master's thesis, Laboratory of Computer and Information Science, Helsinki University of Technology.

Viitaniemi, V. and Laaksonen, J. (2002). Browsing an electronic mail-order catalogue with PicSOM content-based image retrieval system, *Proceedings of 10th Finnish Artificial Intelligence Conference (STEP 2002)*, Oulu, Finland, pp. 170–181.

Vleugels, J. and Veltkamp, R. C. (2002). Efficient image retrieval through vantage objects, *Pattern Recognition* **35**(1): 69–80.

Volmer, S. (2002). Fast approximate nearest-neighbor queries in metric feature spaces by buoy indexing, *Proceedings of 5th International Conference on Visual Information System*, HsinChu, Taiwan, pp. 36–49.

W3C (2003). The W3C Semantic Web home page, http://www.w3.org/2001/sw/.

Wang, J. Z., Liu, J. and Wiederhold, G. (2001). SIMPLIcity: Semantics-sensitive integrated matching for picture libraries, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **23**(9): 947–963.

Weber, R., Schek, H.-J. and Blott, S. (1998). A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces, *Proceedings of 24rd International Conference on Very Large Data Bases (VLDB'98)*, New York City, NY, USA, pp. 194–205.

White, D. A. and Jain, R. (1996a). Similarity indexing: Algorithms and performance, *in* I. K. Sethi and R. J. Jain (eds), *Storage and Retrieval for Image and Video Databases IV*, Vol. 2670 of *Proceedings of SPIE*, pp. 62–73.

White, D. A. and Jain, R. (1996b). Similarity indexing with the SS-tree, *Proceedings of 12th IEEE International Conference on Data Engineering*, New Orleans, LA, USA, pp. 516–523.

Wood, M. E. J., Campbell, N. W. and Thomas, B. T. (1998). Interative refinement by relevance feedback in content-based digital image retrieval, *Proceedings of 6th International ACM Multimedia Conference (ACM MM '98)*, Bristol, UK, pp. 13–20.

Wu, H., Lu, H. and Ma, S. (2002). The role of sample distribution in relevance feedback for content based image retrieval, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2002)*, Vol. 1, Lausanne, Switzerland, pp. 225–228.

Wu, P. and Manjunath, B. S. (2001). Adaptive nearest neighbor search for relevance feedback in large image databases, *Proceedings of 9th ACM International Conference on Multimedia (ACM MM '01)*, Ottawa, Canada, pp. 89–97.

Wu, P., Manjunath, B. S. and Shin, H. D. (2000a). Dimensionality reduction for image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2000)*, Vol. 3, Vancouver, Canada, pp. 726–729.

Wu, Y. and Zhang, A. (2002). A feature re-weighting approach for relevance feedback in image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2002)*, Vol. 2, Rochester, NY, USA, pp. 581–584.

Wu, Y., Tian, Q. and Huang, T. S. (2000b). Discriminant-EM algorithm with application to image retrieval, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2000)*, Vol. 1, Hilton Head Island, SC, USA, pp. 222–227.

Yang, Z., Wang, X. and Kuo, C.-C. J. (1998). Interactive image retrieval: Concept, procedure and tools, *Proceedings of the 32nd Asilomar Conference on Signals, Systems & Computers*, Vol. 1, Pacific Grove, Canada, pp. 261–265.

Ye, H. and Xu, G. (2003). Fast search in large-scale image database using vector quantization, *Proceedings of International Conference on Image and Video Retrieval (CIVR 2003)*, Urbana, IL, USA, pp. 477–487.

Yee, K.-P., Swearingen, K., Li, K. and Hearst, M. (2003). Faceted metadata for image search and browsing, *Proceedings of ACM Conference on Human Factors in Computing Systems (ACM CHI 2003)*, Fort Lauderdale, FL, USA, pp. 401–408.

Yin, P.-Y. and Yeh, C.-C. (2002). Content-based retrieval from trademark databases, *Pattern Recognition Letters* **23**(1-3): 113–126.

Yoo, H.-W., Jung, S.-H., Jang, D.-S. and Na, Y.-K. (2002). Extraction of major object features using VQ clustering for content-based image retrieval, *Pattern Recognition* **35**(5): 1115–1126.

Yoon, J. and Jayant, N. (2001). Relevance feedback for semantics based image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 1, Thessaloniki, Greece, pp. 42–45.

Yoshitaka, A. and Ichikawa, T. (1999). A survey of content-based retrieval for multimedia databases, *IEEE Transactions on Knowledge and Data Engineering* **11**(1): 81–93.

Yu, D. and Zhang, A. (2000). ClusterTree: Integration of cluster representation and nearest neighbor search for image databases, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, Vol. 3, New York City, NY, USA, pp. 1713–1716.

Zahn, C. T. and Roskies, R. Z. (1972). Fourier descriptors for plane closed curves, *IEEE Transactions on Computers* **C-21**(3): 269–281.

Zhang, C. and Chen, T. (2002). An active learning framework for content-based information retrieval, *IEEE Transactions on Multimedia* **4**(2): 260–268.

Zhang, H. and Zhong, D. (1995). A scheme for visual feature based image indexing, *Storage and Retrieval for Image and Video Databases III (SPIE)*, Vol. 2420 of *SPIE Proceedings Series*, San Jose, CA, USA.

Zhang, L., Lin, F. and Zhang, B. (2001). Support vector machine learning for image retrieval, *Proceedings of IEEE International Conference on Image Processing (ICIP 2001)*, Vol. 2, Thessaloniki, Greece, pp. 721–724.

Zhao, R. and Grotsky, W. I. (2002). Narrowing the semantic gap—improved text-based web document retrieval using visual features, *IEEE Transactions on Multimedia* **4**(2): 189–200.

Zhou, X. S. and Huang, T. S. (2002). Unifying keywords and visual contents in image retrieval, *IEEE Multimedia* **9**(2): 23–33.

Zhou, X. S. and Huang, T. S. (2003). Relevance feedback for image retrieval: A comprehensive review, *Multimedia Systems* **8**(6): 536–544.

Zhu, L. and Zhang, A. (2000). Supporting multi-example image queries in image databases, *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000)*, Vol. 2, New York City, NY, USA, pp. 697–700.