Centrum voor Wiskunde en Informatica

REPORTRAPPORT

PNA

Probability, Networks and Algorithms

Probability, Networks and Algorithms

Indexing, learning and content-based retrieval for special purpose image databases

M.J. Huiskes, E.J. Pauwels

REPORT PNA-E0423 DECEMBER 2004

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

ISSN 1386-3711

Indexing, learning and content-based retrieval for special purpose image databases

ABSTRACT
This chapter deals with content-based image retrieval in special purpose image databases. As image data is amassed ever more effortlessly, building efficient systems for searching and browsing of image databases becomes increasingly urgent. We provide an overview of the current state-of-the art by taking a tour along the entire "image retrieval chain" – from processing raw image data, through various methods of machine learning, to the interactive presentation of query results. As the key to building successful image retrieval systems is in the detailed and accurate description of the images, we start out with a discussion on content representation and indexing. We describe various methods to obtain image features, and also introduce the representation of content by means of MPEG-7 metadata. With regard to the search system itself, we focus particularly on interfaces and learning algorithms which facilitate relevance feedback, i.e. on systems that allow for natural interaction with the user in refining queries directly in terms of example images. To this end the literature on this subject is reviewed, and an outline is provided of the special structure of the relevance feedback learning problem. Finally we present a probabilistic approach to relevance feedback that addresses this special structure.

2000 Mathematics Subject Classification: 68T10, 68Q32, 68T45
1998 ACM Computing Classification System: H.3.3, I.2.6, I.4.7, I.4.6, H.5.2
Keywords and Phrases: content-based image retrieval; machine learning; relevance feedback; MPEG-7; feature selection; user interfaces

Indexing, Learning and Content-based Retrieval for Special Purpose Image Databases

Mark J. Huiskes
Centre for Mathematics and Computer Science
Kruislaan 413, 1098SJ Amsterdam, The Netherlands
Mark.Huiskes@cwi.nl

Eric J. Pauwels
Centre for Mathematics and Computer Science
Eric.Pauwels@cwi.nl

Abstract

This chapter deals with content-based image retrieval in special purpose image databases. As image data is amassed ever more effortlessly, building efficient systems for searching and browsing of image databases becomes increasingly urgent. We provide an overview of the current state-of-the art by taking a tour along the entire "image retrieval chain" – from processing raw image data, through various methods of machine learning, to the interactive presentation of query results.

As the key to building successful image retrieval systems is in the detailed and accurate description of the images, we start out with a discussion on content representation and indexing. We describe various methods to obtain image features, and also introduce the representation of content by means of MPEG-7 metadata.

With regard to the search system itself, we focus particularly on interfaces and learning algorithms which facilitate relevance feedback, i.e. on systems that allow for natural interaction with the user in refining queries directly in terms of example images. To this end the literature on this subject is reviewed, and an outline is provided of the special structure of the relevance feedback learning problem. Finally we present a probabilistic approach to relevance feedback that addresses this special structure.

Contents

1 Introduction

In this chapter we are concerned with *content-based* retrieval in *special purpose* image databases. In this context, “content-based” means that we aim to characterize images primarily by analyzing their intrinsic visual content by machine rather than by relying on “external” descriptions. So we let our systems derive descriptions based on the analysis of the image itself, instead of using manually annotated keywords, or as in the case of for instance Google’s image search, using the image caption or text on the webpage adjacent to the image. With “special purpose” we indicate that we restrict ourselves to domains where queries are limited in the terms by which they are formulated and specialized domain knowledge can help us in modeling these terms. Particular examples of such databases occur in domains such as forensics (e.g. fingerprints or mug shot databases), trademark protection (e.g. Eakins et al., 1998), medicine, biomonitoring (e.g. Ranguelova et al., 2004), and interior design (discussed below).

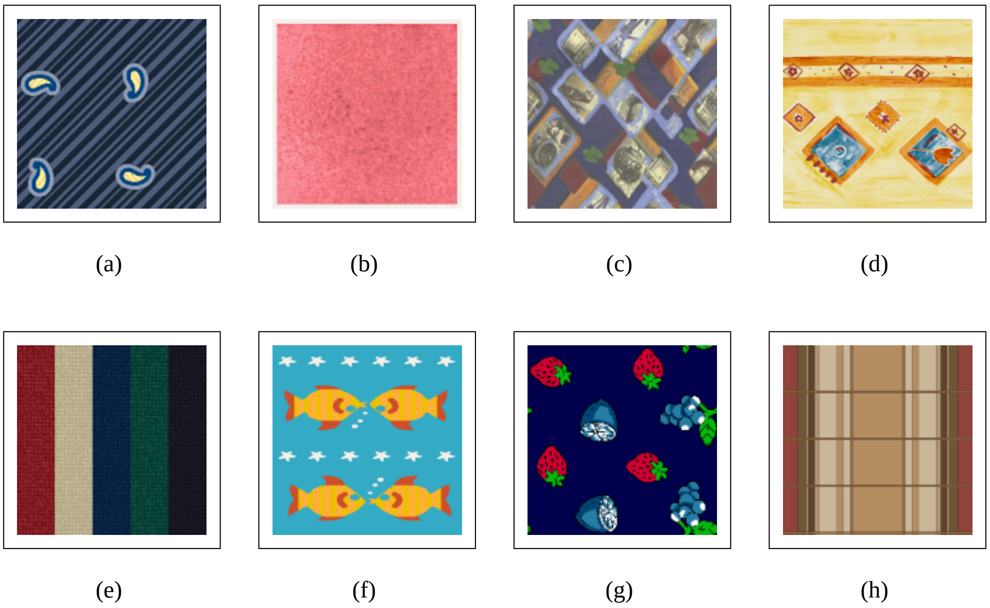
To realize the promise of content-based browsing and searching in image databases, the main obstacle remains the well-known semantic gap. In Smeulders et al. (2000) it is defined as “the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation". The automatically obtained visual measures, or *features*, are typically low-level and often fall short of the semantics of human subjectivity.

Additionally, we must deal with the fact that people often have different interpretations of the same image, or worse, that one and the same person has different perceptions in different situations.

Relevance feedback has often been suggested as a (partial) solution to these formidable problems, in particular to dealing with the user- and task-dependence of image interpretation. The most natural way to provide such feedback is, arguably, by letting the user select both positive and negative examples to indicate his respective preferences and dislikes. This allows the system to extract which features are important for the query at hand.

Even when using relevance feedback, however, ultimate success will still depend on the richness and accuracy of the *representations* we construct of our images. In this chapter we will discuss both the construction of such representations, as a combination of modeling and learning, and the inference techniques to use the representations in transforming the feedback into image relevance measures. Machine learning will play an important role throughout, both for building a higher level understanding of the designs using lower-level building blocks, and in the task of identifying implied preferences from user feedback.

The methods we discuss here will to a large extent apply to the development of content-based retrieval systems for any type of specialized image database. As a real-life example of a system we describe one for the subdomain of decoration design images.


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 1. Examples of design images as used in the textile industry.

Decoration design images form a class of images consisting of patterns used in, for instance, various types of textile products (clothes, curtains, carpets) and wallpaper. Figure 1 shows a number of typical examples. Because of the widespread use of images in the textile, fashion and interior decoration industries, the development of retrieval methods for this economically important domain is also valuable in its own right.

As an introduction, we will take a short tour along the main components of the content-based retrieval system developed in the FOUNDIT project (Pauwels et al., 2003).

1.1 The FOUNDIT CBIR system

The European IST project FOUNDIT aims to develop content-based retrieval systems that are particularly geared towards requirements for searching and browsing digital decoration designs. The definition of requirements of the search system has taken place in close collaboration with the textile industry. Scenarios of use include both customer browsing through design collections, and expert search by designers for, among others, re-purposing.

A guiding principle in the design of the search system has been that the only way to elucidate the user's subjective appreciation and preferences, is by continuously soliciting his or her feedback. This feedback is then harnessed to estimate for each image in the database the likelihood of its relevance with respect to the user's goals whereupon the most promising candidates are displayed for further inspection and feedback.

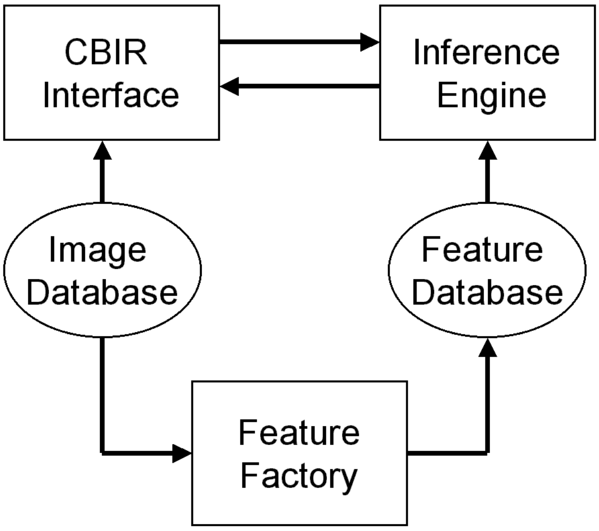


Figure 2. Main architecture of the FOUNDIT system for content-based image retrieval.

Figure 2 illustrates the main components of the system:

- The *graphical user interface* displays a selection of images from the image database and allows the user to provide the system with relevance feedback by selecting examples and counterexamples.
- The *inference engine* transforms this qualitative feedback into a probabilistic relevance measure for each image by analyzing it in terms of the image features.
- The *feature extraction engine*, or feature factory, generates a feature database of visual characteristics by analyzing the content of the images. Unlike the previous components the feature extraction engine operates off-line.

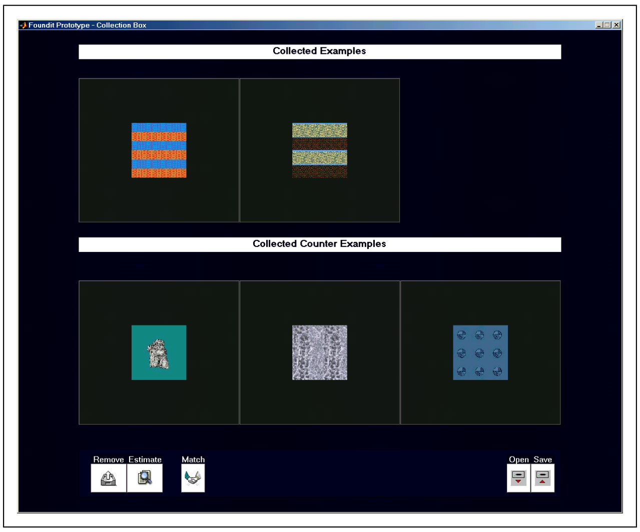

Foundit Prototype - Collection Box
Collected Examples
Collected Counter Examples
Remove
Estimate
Match
Open
Save

Figure 3. Collection box of positive and negative examples. In this example the user has selected two images with horizontal stripes as positive examples (top row). Additionally 3 images were selected as negative examples (bottom row).

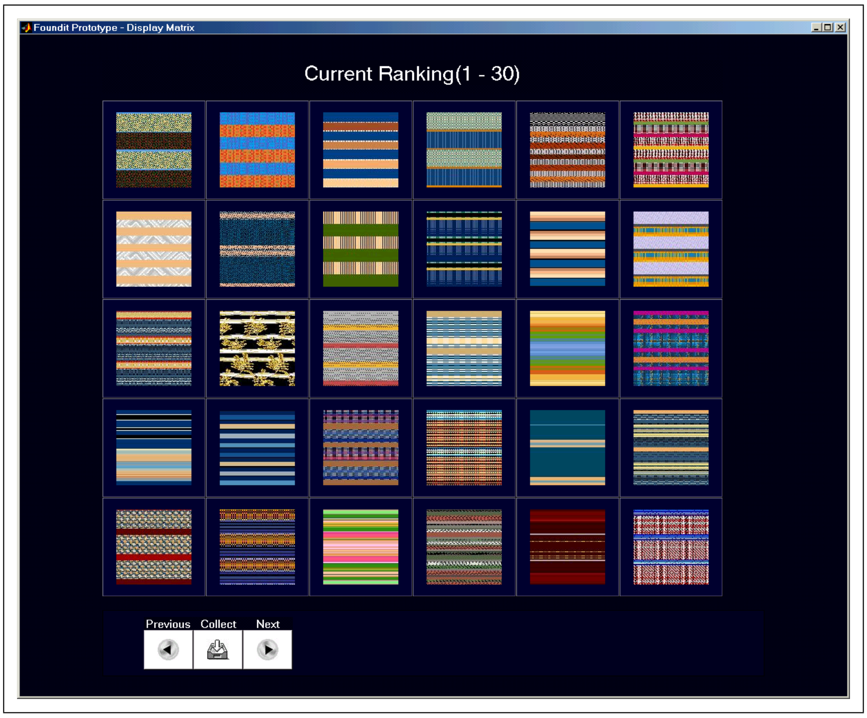
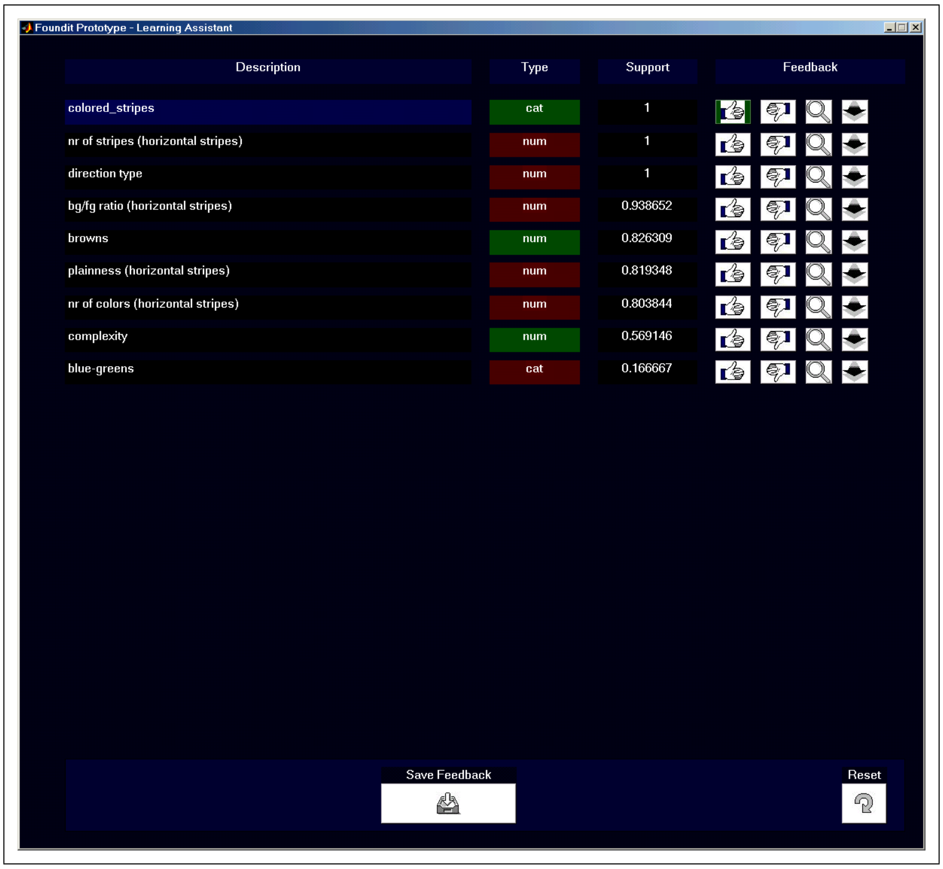


Figure 4. Relevance ranking of database image based on inference engine analysis. Shown are the 30 top ranked database designs that were found based on the collection box of Figure 3. As expected the selection consists of horizontal stripes; subsequent relevance feedback can now be used for searching on additional properties such as color and number of stripes.

A typical query then starts by displaying a random selection of images from the database. The user indicates his preferences by clicking the images, once for images he finds relevant, twice for images that are very dissimilar to what he has in mind. These choices are collected in separate bins of the so-called collection box, as illustrated in Figure 3. Note that for images on which the user has no strong opinion one way or the other, no feedback is provided.

Next, the inference engine is put to work to determine those features or feature combinations that best explain the feedback given. Using a relevance model, this then leads to a ranking of the database images by their predicted relevance which may be used to display the next selection of images (Figure 4.)

In the next cycle the user can add further example images to the collection box, or if desired, remove example images if this better represents his wishes. Additionally we have experimented with showing a list of the most promising individual features in determining relevance, allowing for a very direct type of feedback in terms of feature names. This type of feedback is expected to be useful mainly for expert users. See Figure 5.



Description	Type	Support	Feedback
colored_stripes	cat	1	
nr of stripes (horizontal stripes)	num	1	
direction type	num	1	
bg/fg ratio (horizontal stripes)	num	0.938652	
browns	num	0.826309	
plainness (horizontal stripes)	num	0.819348	
nr of colors (horizontal stripes)	num	0.803844	
complexity	num	0.569146	
blue-greens	cat	0.166667	

Figure 5. Screenshot of the *learning assistant* interface. On the left the system displays a list of features the inference engine considers particularly important in view of the feedback provided by the user. The user is now free to confirm or reject each of these suggestions.

1.2 Outline of the chapter

The remaining sections of this chapter are largely independent and can be read according the reader's interest:

Section 2 Presents an overview of methods for feature extraction and the representation of image content.

Section 3 Discusses the detection of salient elements in design images.

Section 4 Provides a short introduction to the MPEG-7 standard for multimedia content description, in particular in the context of design images.

Section 5 Gives an overview of inference and learning techniques for relevance feedback, and focuses on one method specifically aimed at the special structure of feedback data.

Section 6 Conclusion and outlook.

2 Representation of image content: feature extraction

2.1 Introduction

As indicated above the rich and accurate representation of image content is crucial to the success of any search system. If representation of certain image features is neglected, generally no searching with respect to such feature will be possible other than through accidental correlations with other features.

As generation of features on the fly is generally computationally not feasible with current technology, image representation by features is predominantly a one way street: the full image content is reduced to a single set of fixed, rather inflexible, features. Admittedly, some combination of these features into new features is possible (see for example Minka and Picard, 1997), but this can help us only to a limited extent, i.e. choosing a fixed set of initial features, irrevocably leads to a substantial loss of information. This is in stark contrast to how humans may interpret and re-interpret images depending on different contexts.

Figure 6 provides an illustration of a typical discrepancy between low-level features and high-level perception occurring in the domain of decoration design images.

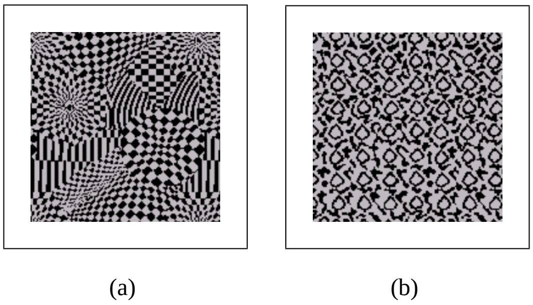


Figure 6. Illustration of the semantic gap for design images. Both image pairs are similar in terms of low-level features, particularly in their color distributions. However, expert designers do not perceive the designs as similar as they are from different "design classes": they would classify image (a) as "optical" and (b) as "texture".

Another issue in feature extraction is its, often unexpectedly, high level of difficulty. Even in cases where we have sufficient domain knowledge to model all or most features that are expected to be of importance in a search, the abundance of possible variations and special cases we may encounter is often rather stunning. In particular the idiosyncracies of the human attentional system are a great source of problems and ambiguities. To give a simple example: an image could consist almost entirely of small diagonal stripes whereas its predominant perception is horizontal, e.g. the stripes may be grouped in such a way they form a horizontal bar in the foreground. It is, however, of paramount importance that features measure characteristics that are perceptually relevant to the user.

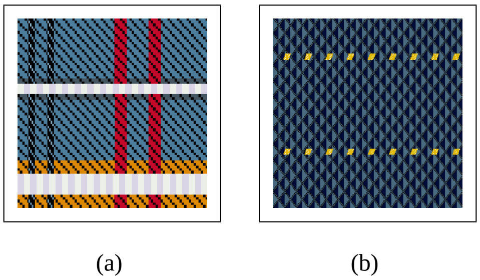


Figure 7. Examples of design images with conflicting direction features and directional perception. In (a) we see strong perceptual evidence for horizontal (and to lesser extent, vertical) lines, whereas strictly numerically the diagonal lines outnumber these by far. Similarly, in (b) we have many diagonal lines and hardly have any horizontal lines or edges at all; still our perception of this design is predominantly horizontal because of the arrangement of the small motifs.

Also note that as an erroneous representation generally leads to correspondingly erroneous behavior in the search system, there is a great need for feature extraction methods that have at least some sort of self- or cross-checking included, such that uncertain cases may be treated as such by the search system or, alternatively, be presented for additional supervised evaluation.

We have found it useful to make a distinction between features based on the overall appearance of an image, and features based on the elements occurring in the image. The former will be discussed in section 2.2, the latter in section 2.3. The detection of design elements is a topic in itself that will be treated in section 3. Organization of feature data by means of the MPEG-7 metadata is discussed in 4.

2.2 Global characterization

We start the overview of design image characterization by a discussion of features designed to capture the overall appearance of an image. We expect the features described here to be useful for general images as well. Features are described for overall appearance with respect to color, texture, complexity and periodicity. Most of the features considered may also be used to describe the appearance of a single image region.

2.2.1 Color

Color is a powerful visual attribute in the perception of similarity between images and, as such, has often served as a key source of features in image retrieval systems (e.g. Flickner et al. (1995), Cox et al. (2000), Swain and Ballard (1991)). Many features have been described in literature for the characterization of color distribution and spatial layout. For the characterization of the global color appearance the challenge lies in the extraction of the perceptually important colors based on their relative occurrence and spatial interactions. See for instance Mojsilovic et al. (2002) and Pass et al. (1996). Also for decoration designs color may play an important role in determining relevance in a search, both with respect to dominant impression and very specific structural wishes (e.g. a customer wishes the motifs in a certain color). We must note however that in many retrieval applications, it is relatively simple to re-color designs after retrieval (e.g. designs are represented in pseudo-color, and color maps may be modified). For expert users color is thus often of little concern.

For the FOUNDIT system we relied primarily on simple code book histogram approaches. Both data-driven and fixed code books were used. For the data-driven code book pixels are sampled from images in the database and K-means clustering is used to determine a set of representative colors. For the fixed code book we use a list of named colors and associated RGB values covering most of the colors generally used in the decorative designs.

For both code book types a pixel is assigned either fully or partially to a bin weighted by its distance to the color associated with the bin. To determine distances between colors we use the Euclidian metric in CIE Lab space. The features consist of the relative contributions of the image to each bin, using thresholds to prevent the contribution of perceptually irrelevant colors.

Additional features are obtained by adding contributions of color bins that can be associated with a single color name, e.g. all blue-ish colors. These metacolor features show very satisfactory performance with regard to capturing subjective experience of dominant color.

In addition we used a number of the color descriptors defined in the MPEG-7 standard, described in detail in Ohm et al. (2002). For the decoration design images we found the following descriptors to be useful:

- the Dominant Color Descriptor (DCD), which provides a very compact description of the dominant colors in an image. Unlike the traditional histogram-based methods, this descriptor computes a small sequence of representative colors in the image with associated area percentages and measures of color variance. Additionally an overall spatial homogeneity coefficient of the dominant colors in the image is computed.

- the Scalable Color Descriptor (SCD), which applies a Haar transform-based encoding scheme across values of a color histogram in HSV color space.

- the Color Structure Descriptor (CSD), which takes into account not only the color distribution of the image, but also the local spatial structure of the colors.

Finally, simple features based on saturation and value from the HSV color space are used as well since we found they correlate well with the experienced degree of “liveliness” and brightness.

2.2.2 Direction and texture

In design images global and local directions of design patterns are often central to the design, think for instance of striped patterns, or tartans. It is thus important to be able to identify those orientations in an image that are perceived as dominant.

Many methods are available to assess typical line or stripe pattern angles (e.g. Freeman and Adelson (1991), Bigün and Granlund (1987), Kass and Witkin (1987)). In the FOUNDIT system we have implemented a method based on the Radon transform of an edge map for detection of horizontal, vertical and diagonal dominant directions. Additionally we use pooled bins of the MPEG-7 Edge Histogram Descriptor. The first method counts edges occurring on lines of certain orientation, the second method is complementary in the sense that it counts edges of certain orientation. Note that these simple features often fail for curves that are not sufficiently straight, e.g. for patterns of horizontal waves. For such cases we must first detect the respective design elements after which we may establish their orientation. Another issue that supports this latter approach is that the features often react to edges that are not relevant, e.g. a design with a vertically oriented texture ground with a horizontal stripe, will often be quantified as predominantly vertical whereas subjective experience would usually favor a horizontal interpretation (remember figure 7 of section 2).

The occurrence of stripes in designs is so common that we have developed a template-based method for their detection. This method is based both on grouping of edges and evaluation of homogeneity within the stripe.

A stripe is considered to constitute a region in an image which (i) is bounded by two relatively straight lines spanning the image, and (ii) has a relative homogeneous appearance in between those lines, which differs from the appearance outside these lines. The sense in which the homogeneity is to be understood is not defined in the algorithm. In practice this means we assume that the user provides an indexed image in which each index represents a certain homogeneity type. One can think for instance of an image resulting from a color or texture segmentation (or from a combination of both).

Several aspects with regard to stripes may play a role in perceived similarity of designs. Think of the occurrence and location of stripes, their orientation, the width of the stripes, variation in the stripe widths and color and texture of the stripes. Useful features quantifying such properties are extracted at various stages of the algorithm. For instance after the edge line detection, we count the number of edge lines in an image, or consider the relative number of edge lines. Once stripes have been detected, we compute the density of the stripes in the image; additionally, we compute the average distance between the stripes.

A number of composite features were computed as well, e.g. a feature measuring the occurrence of both horizontal and vertical or two diagonal directions of(thin) stripes simultaneously, for the detection of tartan images; and features assessing the possibility that the background of an image consists of stripes.

Many methods are also available for characterizing image texture (e.g. Reed and Du Buf (1993), Bovik et al. (1990), Manjunath and Ma (1996), Liu and Picard (1996), Randen and Husoy (1999), Mao and Jain (1992), Gimel'farb and Jain (1996), Gotlieb and Kreyszig (1990), Laine and Fan (1993), Lin et al. (1997a)). From the MPEG-7 texture descriptors (see Choi et al., 2002), we have used the following descriptors:

- the Homogeneous Texture Descriptor (HTD), which characterizes texture using the mean energy and the energy deviation from a set of 30 frequency channels. The frequency plane partitioning is uniform along the angular direction (equal steps of 30 degrees), but not uniform along the radial direction (which is on an octave scale). There is some evidence that the early visual processing in the human visual cortex can be modelled well using a similar frequency layout.

- the Texture Browsing Descriptor (TBD), which specifies the perceptual characterization of a texture in terms of regularity (4 levels), coarseness (2 quantized scales) and directionality (2 quantized directions). The filtering is performed using a Gabor filter extraction method with a similar frequency layout as in the HTD.

- the Edge Histogram Descriptor (EHD), which describes the spatial distribution of the edges in an image.

2.2.3 Complexity

For designers overall complexity is an influential factor in the perception of design similarity. Of course, complexity is not a well-defined quantity that can be determined objectively. Rather we observe that a number of design features correlate well with subjectively perceived design complexity, e.g. the number of colors occurring in the design, the "amount of discontinuity", its "crowdedness" or its "level of detail".

We modeled the level of detail by means of a summarizing feature resulting from a multiresolution analysis, along the lines of for instance Boggess and Narcowich (2001), where the image is decomposed into terms representing contributions at different levels of scale.

To this end we use a grayscale image of the original (with 64 levels). Its wavelet decomposition using the Daubechies-6 wavelet family type is computed and features are determined by taking the maximum absolute deviation of the coefficients for the approximation and the levels of detail (horizontal, vertical and diagonal) at 4 different scales. The summarizing quantity takes a ratio of energy in the lower scales to energy in the higher scales.

A reasonable correlation with perceived overall complexity was obtained by taking a weighted sum of this quantity and the total number of edges found by the Edge Histogram Descriptor (section 2.2.2) and the number of colors in the image. The weights were chosen to optimize performance on an annotated test set.

2.2.4 Periodicity

Many images contain patterns which are periodic, i.e. patterns that are invariant to certain translations. In images we may have periodicity in one direction (so-called “friezes”) and periodicity in two independent directions (so-called “wallpapers”.) The repeating elements can be extracted by means of autocorrelation analysis or Fourier approaches. See for instance Lin et al. (1997b) and Russ (1995).

General symmetries (i.e. rigid transformations that keep the pattern invariant) of the plane are compositions of translations, rotations and reflections. In Schattschneider (1978) a detailed analysis of plane isometries and plane symmetry groups is presented. It is shown that for two dimensional designs there are 7 frieze groups describing patterns that repeat along one direction and 17 wallpaper groups for patterns that repeat along two linearly independent directions to tile the plane. The groups vary in which additional symmetry types are present. Liu et al. (2004) provides a computational model for periodic pattern classification based on these groups. Using this model one can identify the symmetry group of the pattern and extract a representative motif.

For extracting the translation lattice, we have used an autocorrelation approach along the lines of Lin et al. (1997b). The maxima of the autocorrelation function give a set of candidate lattice points. The task is then to find the shortest linearly independent translation vectors that generate this lattice. For the frieze patterns this is a single vector, for the wallpaper patterns two vectors are needed. Liu et al. (2004) introduce a method based regions of dominance to robustly determine these translation vectors.

An additional method to investigate periodicity is by extracting the motifs by means of figure-ground segregation, followed by an analysis of their arrangement and variation (see section 2.3.3).

2.3 Representation of region properties and relations

Once design elements have been identified, more detailed characterization of designs becomes possible. Procedures to determine such elements are discussed in section 3.

2.3.1 Region properties

For a given connected image region, representing for instance a design motif, we may compute various elementary properties. We mention for example: size (e.g. relative area, extent, equivalent diameter, perimeter, length of major and minor axis); orientation (based on for instance bounding box or fitted ellipse); eccentricity (or elongation, cirularity, rectangularity);

convexity (or solidity, compactness); color and texture; central and invariant moments; curvature (e.g. total absolute curvature, bending energy); contrast with ground; number of holes; fractal dimension (rate at which the perimeter of an object increases as the measurement scale is reduced). For definitions and formulas of these properties we refer to elementary textbooks on image processing such as Russ (1995) and Pratt (1991). When a design consists of only one type of motif, we may use these quantities directly as features. In case there are different types of motifs, we detect common properties and properties that show variation.

Of course, similar properties may also be computed for the design ground. Particularly interesting here is ground texture that may correlate with various kind of textile treatments, such as batik, chiné, dots and fine lines. To this end we may use the general texture features discussed in section 2.2.2.

2.3.2 Shape

Many of the region properties discussed before are related to region shape. The property of shape is very active subject of research and warrants some further discussion. Next to the mentioned simple properties, various shape similarity metrics with associated feature space have been developed. See for instance Veltkamp and Hagedoorn (2001) and Bober et al. (2002) for an overview.

We have used the MPEG-7 visual shape descriptors (see Bober et al. (2002)):

- the Region Shape Descriptor (RSD), which quantifies the pixel distribution within a 2-D object or region. It is based on both boundary and internal pictures, and it can describe complex objects consisting of multiple disconnected regions as well as simple objects with or without holes. It uses a complex angular radial transform (ART) defined on a unit disk in polar coordinates.

- the Contour Shape Descriptor (CSD), which is based on the curvature scale space (CSS) representation of the contour.

Based on the shape shape descriptors we can construct shape features measuring membership to certain shape categories. A few example categories are shown in Figure 8.

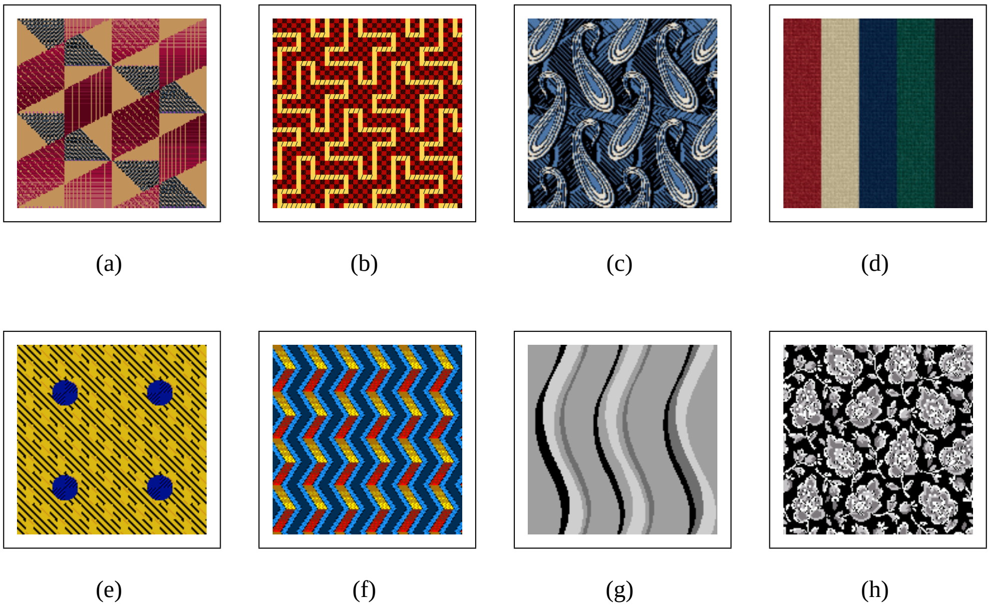


Figure 8. Examples of shape categories important for designs: (a) diamonds, (b) pied-de-poule, (c) paisley, (d) stripe, (e) pois (circles), (f) chevron, (g) movement (waves), (h) leaves

For the category modeling we have taken a case-based approach using exemplars, which is a convenient method to develop useful features from high dimensional feature spaces.

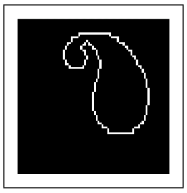
For each shape category we have taken a number of representative cases (e.g. see Figure 9 (a)). These are called the exemplars; the shape category feature a given shape is then based on the minimum distance to the exemplars shape. Figure 9 (b) shows distances for a few shapes to the paisley examplar of Figure 9 (a) based on the contour shape descriptor.

2.3.3 Object variation and spatial organization

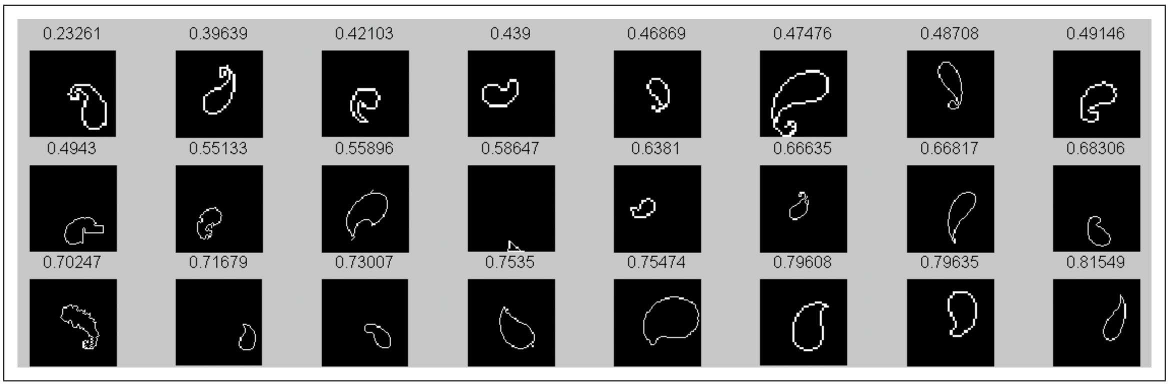
Once the salient elements in an image have been determined it becomes interesting not only to look at their intrinsic properties, but also to consider the relations between such elements. In particular, we are interested in the similarities and differences between the various elements, and in their spatial organization.

Variation between the intrinsic properties can be measured directly, e.g. by using the median absolute deviation (MAD). Using such features we can detect, for instance, if the motifs in a design possess similar shape but differ in orientation.

Further features are based on the distances between the elements (design spacing) and measure if and how they are organized in a grid pattern.



(a)



(b)

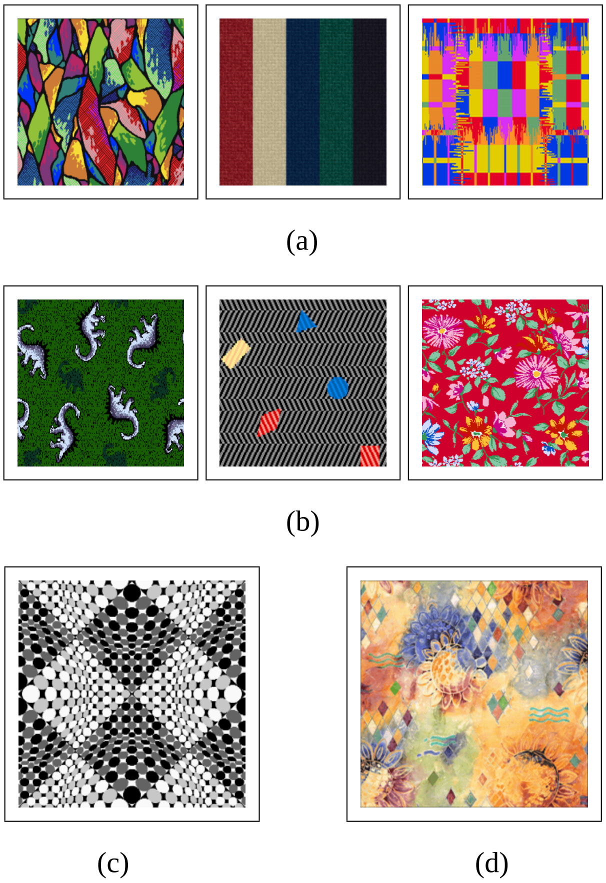
Figure 9. (a) Exemplar contour for the paisley shape; (b) Contour distance to the paisley exemplar shape as measured by the MPEG-7 Contour Shape Descriptor.

3 Detection of salient design image elements by figure-ground segregation

As mentioned the performance of the CBIR system relies to an important extent on the quality of the chosen design representations. Additionally we have seen that for meaningful searching and browsing through design collections higher-level characterizations based on the individual elements in the design are essential. Only then, if such elements can be identified, it becomes feasible to quantify visual properties such as shape, spatial pattern and organization, and variation in for instance color, shape and orientation among the elements.

In an important subset of the decoration designs the identification of individual design elements can take place by figure-ground segregation. The definition of the figure-ground segregation problem is, however, by no means trivial. Design elements may be arranged in a large variety of ways: they may be overlaid, may fade over into each other, or may form tilings of the image plane. Furthermore the elements may take part in many types of spatial patterns and groupings. Within such arrangements the design elements vary in their level of *salience*, i.e. by the extent to which ‘they stand out’. For figure-ground segregation we are interested in those cases where design elements are arranged on a ground, i.e. the case where a number of, usually isolated, salient elements stand out on a non-salient ground. Clearly not all designs possess such ground structure, see for instance Figure 1 (a).

Occurrence of ground structure is often not clear due to the situation that the ordering among the design elements in terms of their salience is not clear. In other cases several figure-ground interpretations are possible simultaneously. An example is shown in Figure 10 (c). The image can be interpreted as black on white, or as white on black. In some cases the occurrence of ground structure is clear, but it is still hard to determine the ground accurately (Figure 10 (d)). The latter two effects are referred to as ground *instability*.



(a)

(b)

(c)

(d)

Figure 10. Examples of design images: (a) images without background; (b) images with background; (c) and (d) images for which the background-foreground structure is unclear.

As a final issue we mention the occurrence of nested grounds. An example to this effect is shown in Figure 11 (a). The image can be interpreted to consist of three layers: a plain green layer, a layer of heart motifs and four angels. The background can thus be either the plain layer, or this layer together with the hearts. A special case of this problem occurs in relation to designs consisting entirely of texture where the entire image may be taken to consist of background. In such cases it is often still useful to analyze the texture additionally in terms of its figure-ground structure, see Figure 11 (b).

In the following we give a concise description of a method for figure-ground segregation based on the identification of salient color-patterns, the color coalitions, which is described in more detail in Huiskes and Pauwels (2003).

Patterns of color and regions of color texture play an important role in the visual structure of decoration designs. Consequently many pixels in design images can be naturally interpreted to take part in various color combinations. As an example consider the design image of Figure 12 (a). The background in this image consists of pixels of two colors: red and black. Rather than viewing such pixels as either red or black it is more natural to view both types of pixels as part of a red-and-black region. Moreover, pixels are often part of a nested sequence of such color combinations. This may be seen by repeating the original design to arrive at the image of Figure 12 (b). The original background pixels are now part of a larger pattern also including the so-called pied-de-poule motifs, which are yellow. Depending upon the scale at which a design

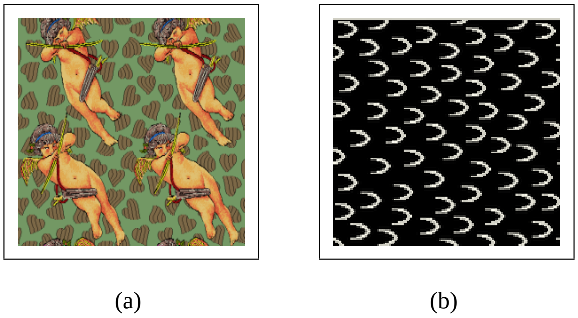


Figure 11. Examples of design images for which the figure-ground structure consists of multiple levels.

is perceived the red and black pixels may thus also take part in a red-black-yellow combination.

We have set out to explore methods for color texture segmentation by direct analysis of the color combinations occurring in an image, i.e. we intend to find the natural color combinations which we shall then call color coalitions.

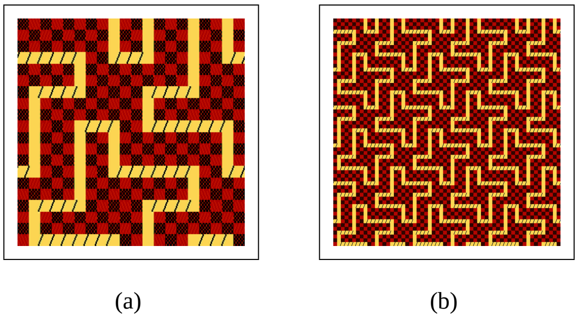


Figure 12. (a) An example design image. (b) The same design image repeated 9 times and resized to its original size.

In section 3.1 we introduce the color coalition labeling and describe its application to the detection of regions of color texture at various scales. Next we discuss its application to figure-ground segregation. We propose a strategy consisting of three main steps:

1. Obtain initial candidates for the background by multi-scale detection of color texture regions (section 3.1).

2. Assess the appropriateness of the individual candidates by an N-nearest neighbor classification algorithm based on visual cues such as relative size, connectedness and massiveness (section 3.2).

3. Integrate the results of the previous steps to produce a hierarchical description of the figure-ground structure of the design (section 3.3).

The algorithms are tested by application to images from two decoration design databases. One is a database of tie designs from an Italian designer company. The other database has been provided by a manufacturer of CAD/CAM systems for the textile industry and includes a wide range of decoration design types.

In section 3.4 we list the results obtained for the two test sets and discuss the general performance of the approach.

3.1 Detection of color coalitions

The task of finding color texture regions is closely related to general image segmentation. As any segmentation problem it is about the grouping of pixels that, in some sense, belong together. The approach we take here is based on direct analysis of the color combinations occurring in the image. As the level of homogeneity of a region depends on the scale under consideration, we must investigate the occurrence of color combinations at various scales. For each scale we then define a color coalition labeling that provides each pixel with a label uniquely identifying the colors occurring in a structuring element around the pixel.

We restrict ourselves to generate candidate regions for the image background and will not attempt a full bottom-up segmentation here. Moreover unlike in most segmentation methods we will not demand texture regions to be connected, nor will we attempt to assign every pixel to a segment region.

The algorithm for the construction of color texture regions for a fixed scale is divided in the following main stages:

1. construct color coalition labeling;
2. erode label image and analyze homogeneity of remaining color combinations;
3. grow the resulting color coalitions into color texture regions.

These stages are outlined in Figure 13 and will be further detailed below.

3.1.1 Color coalition labeling

In the following we consider indexed images where each pixel has an associated integer value that either refers to a color in a colormap or is equal to zero, indicating that the color of the pixel is to be ignored by the algorithm. More formally, we define an image f as a mapping of a subset $\mathcal{D}_f$ of the discrete space $\mathbb{Z}^2$, called the definition domain of the image, into the set of indices:

$$f : \mathcal{D}_f \subset \mathbb{Z}^2 \to \{0\} \cup \mathcal{C}_f = \{0, 1, \ldots, N\}, \tag{1}$$

where $\mathcal{C}_f = \{1, \ldots, N\}$ is the set of color indices of the image. In practice the definition domain is usually a rectangular frame referred to as the image plane of pixels.

For indexed images we define the *index* or *color set* $\mathrm{cs}_c(f)$ of index c as the set of pixels with index c: $\mathrm{cs}_c(f) = \{x | f(x) = c\}$, or as binary image:

$$[\mathrm{cs}_c(f)](x) = \begin{cases} 1 & \text{if} \quad f(x) = c \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

We further define the erosion of an indexed image as the summation of binary erosions performed on the individual color sets while keeping the original indices:

$$\varepsilon_B(f) = \sum_c c \, \varepsilon_B(\mathrm{cs}_c(f)), \tag{3}$$

where B is the structuring element and summation and scalar multiplication are pixel-wise.

For each pixel x we consider the set of colors $\omega_{B_s}(x)$ occurring in a structuring element B_s^x of scale s:

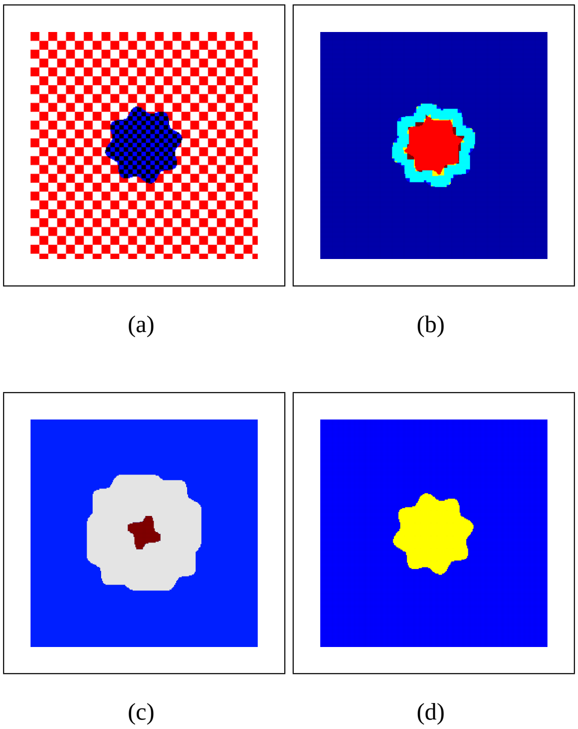


Figure 13. Main stages in construction of color texture regions: (a) test image of 256 by 256 pixels consisting of two regions of color texture; (b) based on a rectangular window of 13 by 13 pixels structuring element the image has 7 distinct color sets; (c) after erosion and homogeneity checking two color sets remain; white pixels in this image have index 0 and do not correspond to a color combination; (d) growing leads to two regions of color texture.

$$\omega_{B_s}(x) = \{c \in \mathcal{C}_f | \exists y \in B_s^x : f(y) = c\}. \tag{4}$$

Each such subset of $\mathcal{C}_f$ is referred to as a *color combination*, and $\omega_{B_s}(x)$ is called the color combination associated with pixel x at scale s. For the structuring element we will usually take a rectangular window with the centre pixel as element origin.

We define the color coalition labeling of f as follows. Let Ω_{B_s} be the set of all color combinations occurring in the image at scale s, then we associate with each combination ω in Ω_{B_s} a label $\lambda_{B_s}(\omega)$ in the order of encounter of such combinations in a forward scan of the image. The color coalition labeling $\Lambda_{B_s}(f)$ of f is then defined by

$$[\Lambda_{B_s}(f)](x) = \lambda_{B_s}(\omega_{B_s}(x)). \tag{5}$$

An example of a color coalition labeling is shown in Figure 13 (b).

3.1.2 Color coalition selection

Our aim is to select the principal color combinations of the image, i.e. those color combinations that are most appropriate to extend to full color texture regions for a given scale. To this end we erode each of the index sets of the color coalition labeling under the tentative assumption that the color combinations occurring at the boundaries of regions of color texture are generally thinner than the interiors of such regions. For the erosion we use a structuring element B_t of scale t, i.e. we construct $\varepsilon_{B_t}(\Lambda_{B_s}(f))$. We denote an eroded set associated with ω by $R(\omega)$, i.e. we take

$$R(\omega) = \varepsilon_{B_t}(\mathrm{cs}_{\lambda_{B_s}(\omega)}). \tag{6}$$

As we are interested in finding regions of homogeneous color texture we further investigate homogeneity statistics for color combinations ω for which $R(\omega)$ is non-empty. Note that if statistics are computed based on a structuring element of scale s, taking $t \geq s$ ensures that colors surrounding a region of color texture cannot affect the homogeneity of the statistics in an eroded color set.

So let $S_{B_s}(x)$ be the local statistics at pixel x taken over pixels in a structuring element B_s, and consider a surviving color combination $\omega : R(\omega) \neq \emptyset$. We accept ω as a color coalition if the following two conditions hold:

1. $R(\omega)$ still contains all colors of the color combination ω.

2. The coefficients of variation of S_{B_s} on $R(\omega)$ are smaller than a given threshold

Both the choice of statistics and of the scale for the erosion structuring element t are subject to a trade-off between the aims of suppression of boundary color combinations and still being able to detect color texture regions that have a relatively large interior[1] scale relative to their exterior scale. We obtained best results by using the erosion as the main mechanism in reducing the number of candidate color combinations (we set $t = 1.5s$), and kept the statistics as simple as possible to allow for maximum detection of color textures. In fact, we take only the relative number of pixels of the dominant color in the structuring element as a statistic. The computation of the coalition labeling and the local statistic can both be implemented taking a single forward scan and a moving histograms approach (see Van Droogenbroeck and Talbot (1996)).

3.1.3 Region growing strategies

Next the color texture regions associated with the principal color combinations are determined by region growing of the eroded color sets. If we denote the final color texture region by $G(\omega) = \mathcal{G}(R(\omega))$ then for a pixel x to be assigned to $G(\omega)$ it should satisfy at least the following conditions:

1. the pixel must have a color index belonging to the color combination: $f(x) \in \omega$.

2. the pixel must have the remaining colors of the color combination in its structuring element: $\omega \subset \omega_{B_s}(x)$.

The pixels in $R(\omega)$ satisfy both conditions; also note that the conditions allow pixels at boundaries of texture regions to have additional colors in their structuring element.

This still leaves the important issue of how to assign pixels for which more than one color combination is feasible. Several strategies are possible such as assigning to the closest or the largest feasible eroded set. In our application we have obtained best results sofar by assigning to the color combination for which the associated eroded region has an average color histogram that is closest to the color histogram of the structuring element of the pixel.

For each scale we thus get a segmentation of the image in regions corresponding to the principal color combinations and a set of pixels with label zero that are not assigned to any color combination.

[1] We define the interior scale of a set as the smallest scale at which a set is homogeneous; the exterior scale as the smallest scale at which the erosion of the set is empty

3.2 Classification

To determine the background quality of color texture regions, we take a simple yet effective approach based on weighted N-nearest neighbor classification. Based on a number of property variables or features of the region the *ground probability* is estimated that the region is suitable to serve as a background region.

Classification takes place by using a training set of sample regions with features $x_i, i = 1, \ldots, n$ that have been assigned a ground probability $p(x_i)$ by manual annotation. The probability $p(x)$ of a region with features x is determined by taking a weighted average of the probabilities of its N nearest neighbors in feature space, see for instance Duda and Hart (1973).

The feature variables were chosen by experimentation with the aim of reaching a high level of consistency and a low level of ambiguity:

- Relative area: the region area relative to the total image area.
- Filling coefficient: background regions often possess a complement consisting of components that are not connected to the border and which are removed after filling the background region (see for instance Soille (1999) for morphological operations such as hole removal). Let X be the background region, X^c its complement and $\bar{X}$ the background region after hole removal, then the filling coefficient $\mathrm{fc}(X)$ is defined as

$$\mathrm{fc}(X) = \begin{cases} 1 - A([\bar{X}]^c)/A(X^c) \quad \text{if} \quad X^c \neq \emptyset \\ 1 \quad \text{if} \quad X^c = \emptyset, \end{cases} \tag{7}$$

  where $A(X)$ is the area in pixels of region X.

- Spatial reach: measures if the region occurs only in certain parts of the image or all over the image; the image is covered by a grid of boxes and spatial reach is measured by counting the relative number of boxes that are occupied by the region.
- Connectedness: The area of the largest connected component of the region relative to the region area (computed after closing with a small structuring element.)
- Massiveness: the median distance of the region pixels to the region boundary.

The N-nearest neighbor approach allows for straightforward evaluation of inconsistencies and ambiguities. Consistency of the samples can be analyzed by comparing the ground probability of a sample region obtained by classification leaving that example out to the probability obtained by manual annotation. Let p_{-i} be the ground probability obtained by classification using all samples except sample i, then we define the consistency for sample i as $\rho_i = |p_{-i} - p(x_i)|$. In our study we took 350 samples of which only 8 had a consistency smaller than 0.75. It is also simple to assess if examples are sufficiently nearby for reliable classification: for instance by comparing distances of examples to new cases to be classified to the average of such distances occurring for the samples in the sample set. If relatively empty regions or problem cases are encountered additional samples may be added.

3.3 Synthesis

Using the color coalition labeling approach of section 3.1 we obtain color coalitions for a sequence of scales. In this study we took 8 scales that were equally distributed over a range from a smallest window of 3 by 3 pixels to a rectangular window of about 30% of the image size. All resulting regions were classified using the method of section 3.2. Each region with a ground probability greater than 0.5 is accepted as a potential ground (although ground probabilities are generally found to be either 0.0 or 1.0). If a color combination is found to be feasible for serving as ground at more than one scale, we take two criteria into account to decide on the most appropriate region: (i) the simplicity of the region; (ii) the number of scales at which the particular region, or a region very similar to that region, is found (scale robustness). For the simplicity measure of the region we have taken, rather ad hoc, the sum of the number of connected regions in the background and the foreground (after opening each with a small structuring element; see Soille (1999)).

Next we further analyze the determined grounds and their associated color combinations. Every pair of combinations is assigned as either: nested, partially overlapping or disjoint. Large disjoint regions often indicate flipping behavior as in Figure 10 (c). Apart from the analysis of such relations that also includes checking the hypothesis that the entire image consists of a single color texture, for every design a highest quality background is determined using the simplicity and robustness criteria. Based on these results each of the images is automatically assigned to one of four distinct categories or *streams*: I: no figure-ground structure; II: figure-ground structure; III: consists entirely of one color texture, which itself possesses figure-ground structure; IV: consists entirely of one color texture, and does not possess further figure-ground structure. Note that such automatic stream assignment allows for data driven feature computations. For example texture features can be computed for the background regions and full texture images, whereas shape features are computed for foreground elements.

3.4 Results

Benchmarking figure-ground segregation algorithms for designs is generally difficult for the reasons sketched in the introduction: even for humans identification of figure-ground structure is often not unambiguous. We thus choose to restrict ourselves to cases where we clearly have a ground or we clearly do not, and check if the algorithm output aligns with human perception for the cases where occurrence of figure-ground structure is clear and stable. This approach recognizes the notion that strict bottom-up processing is generally infeasible, unless some sort of context is assumed: in this case we assume that we are dealing with images where a ground is to be identified.

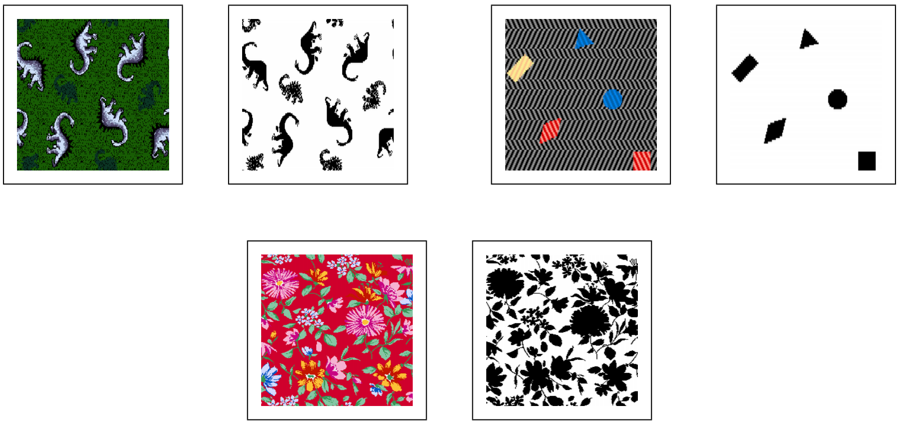


Figure 14. Results of the figure-ground segregation algorithm for the images of Figure 10 (b).

For testing we take three sets of images: (i) Collection 1: 500 images from a specialized database of tie designs, used for training the background detection algorithms; (ii) Collection

2: another 500 images from the same database; (iii) Collection 3: 500 images from a database containing a wide range of decoration designs from the textile industry. As such this database provides a representative test set for images the algorithm is likely to encounter in practice. The images of Collection 2 and 3 have not been used in any way to calibrate the algorithms.

We assigned each of the images in the test sets by manual annotation to either one of the four streams discussed in section 3.3 or to stream V: occurrence of structure not clear or instable. Rates of correct performance are reported in Table 1.

Collection	Stream I	Stream II	Stream III	Stream IV
1	89%	90%	85%	82%
2	92%	91%	77%	90%
3	100%	88%	78%	81%

Table 1. Correct performance rates for images assigned by manual annotation to stream I through IV.

Example images from stream II with results are shown in Figure 14. Errors can largely be attributed to the following types of designs:

- Designs where all colors in the foreground object also occur in the background, and the foreground objects do not disrupt the homogeneity of the background region. An example is shown in Figure 15 (a). Other methods must be used to find such additional structure in the ground.

- Cases where the background consists of a composition of regions, see for instance Figure 15 (b). Currently no combinations of regions are tested for their suitability to serve as ground.

- Cases for which classification is ambiguous, e.g. in images for which the background consists of small isolated patches, that by their shape and layout would rather be expected to be of foreground type. This type of background is hard to detect automatically and generally requires a higher level of design understanding.

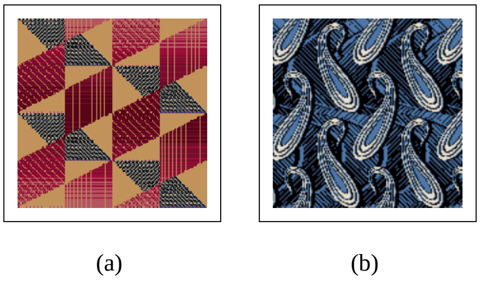
- Cases where the choice of simplicity measure leads inappropriate candidates to be accepted. Occurs very rarely.

- Designs with illumination effects, gradients and special types of noise. Main problem here is the occurrence of noise that is not removed by preprocessing and occurs in only part of a color texture.

- Designs where the interior scale of a background region is large in comparison to its exterior scale. Sometimes the region is not found as a candidate since the color combination region disappears by erosion before it is accepted as homogeneous.

Correct performance is directly related to the occurrence of such types of images in the database. For example the mistakes for Collection 3 are mainly of the first type as the set has a relatively high number of binary images with additional fine structure.

The general conclusion is that the method discussed here works well for a large of decoration design images. There is, however, a class of designs without figure-ground structure that still contain interesting elements relevant to design retrieval. Images in this class include for example geometric mosaics and designs with overlapping elements. Two few examples are shown in Figure 15.



(a) (b)

Figure 15. Examples of design images with (main) elements that cannot be found by figure-ground segregation

Detection of such objects is possible to some extent with general segmentation methods (Freixenet et al. (2002), Pal and Pal (1993), Reed and Du Buf (1993)). These methods lead to partitions of the images in homogeneous regions such that the unions of such regions are not homogeneous. However, we have found that individual segmentation methods are often not able to deliver the regions that are of interest from the design interpretation perspective. Among other reasons, this is due to the intricate interplay of the Gestalt grouping principles (see Wertheimer, 1923, e.g. similarity, proximity, goodness-of-curve). Currently we are working towards increased robustness by exploiting both redundancy (i.e. using results of complementary segmentation approaches) and non-accidentality (i.e. by detecting an unexpectedly high degree of ordering in terms of the Gestalt principles).

4 MPEG-7 description of design images

As we are working towards "content-based" image retrieval it is natural to investigate the potential of the MPEG-7 metadata system for content description.

MPEG-7, formally named "Multimedia Content Description Interface" (MPEG7, 2003), is an ISO/IEC standard for describing various types of multimedia information developed by MPEG (Moving Picture Experts Group). Whereas the MPEG-1 through MPEG-4 standards are aimed at representing the content itself, MPEG-7 represents information about the content: "the bits about the bits" so to speak.

There are many types of audiovisual data content that may have associated MPEG-7 descriptions. These include: still pictures, graphics, 3D models, audio, speech, video, and composition information about how these elements are combined in a multimedia presentation (scenarios). The following is based on Huiskes et al. (2003) and will focus on those parts of the standards that are of interest in the context of retrieval of still pictures in general, and of decoration design images in particular.

In these contexts we found that the main purpose of use can be summarized as twofold:

1. organization of image description data: as discussed in section 2 many types of design features are computed; in this process a large variety of data is generated, consisting not only of the numerical values of the features, but also of intermediate results such as image segmentations for which the computation may be time-consuming and which are therefore worth storing. MPEG-7 descriptions can be used to organize all this data in relation to the structure of the image and to provide bookkeeping with respect to, for instance, algorithm version information and chosen parameter settings.

2. structural and semantical image content description to facilitate the quick and efficient identification of interesting and relevant information

```
<CreationInformation>
    <Creation>
        <Creator>
            <Role><Name xml:lang="en">Main designer</Name></Role>
            <Agent xsi:type="PersonType">
                <Name>
                    <GivenName>Mark</GivenName>
                </Name>
            </Agent>
        </Creator>
        <CreationCoordinates>
            <Location>
                <Name xml:lang="en">Amsterdam Design</Name>
                <Region>nl</Region>
            </Location>
            <Date>
                <TimePoint>2003-03-27</TimePoint>
            </Date>
        </CreationCoordinates>
    </Creation>
</CreationInformation>
```

Figure 16. Snippet of an MPEG-7 description.

This will be explained in more detail below; additionally, we shortly discuss the potential of using the MPEG-7 standard in combination with ontology definitions.

4.1 Structure of MPEG-7 descriptions

MPEG-7 descriptions are defined in terms of *descriptors* and *description schemes*.

A descriptor is a feature representation, i.e. the descriptor defines the syntax and the semantics for the representation of a perceptual feature. Descriptors may be both textual and non-textual and some can be computed automatically whereas others are typically obtained by manual annotation.

Description schemes (DSs) expand on the MPEG-7 descriptors by combining individual descriptors and other description schemes into more complex structures by defining the relationships between its components (which may be both descriptors and description schemes). Description schemes are defined following the MPEG-7 Description Definition Language (DDL), which can also be used to define new description schemes.

In practice one may think of an MPEG-7 description as a (possibly compressed) XML file, and of the description schemes as XML schemas (see for instance Goldfarb and Prescod (2001)). A description always consists of an MPEG-7 top element. It may contain either a partial description consisting of any MPEG-7 description unit desired for a particular application, or a description that is complete in the sense that it follows a pre-defined hierarchy of description units. For an overview of this hierarchy, see Salembier and Smith (2002).

An illustrative snippet from an MPEG-7 description is shown below:

4.2 Description schemes

The following is a concise introduction to the various description schemes provided by the MPEG-7 standard that are most useful for decorative design description.

4.2.1 Content management

MPEG-7 offers description schemes for creation information, usage information and media description. The *CreationInformation DS* provides functionality to describe the creation and production of the design, e.g. a title of the design, its creator, creation locations and dates. The *UsageInformation DS* describes information related to the usage rights management and protection, usage records and financial information.

The *MediaInformation DS* describes the location, and coding and storage format of various possible instances of a described design, e.g. of the master copy and various compressed versions.

Additionally a *DescriptionMetadata DS* is available to provide metadata about the descriptors themselves. It allows for description of the creation of the description, e.g. which algorithm version is used, where is it located and on which parameter settings is it based.

4.2.2 Content structure

The content structure description schemes allow for detailed description of the design structure in terms of its constituent regions and moreover provide a framework for organizing the various types of visual descriptors.

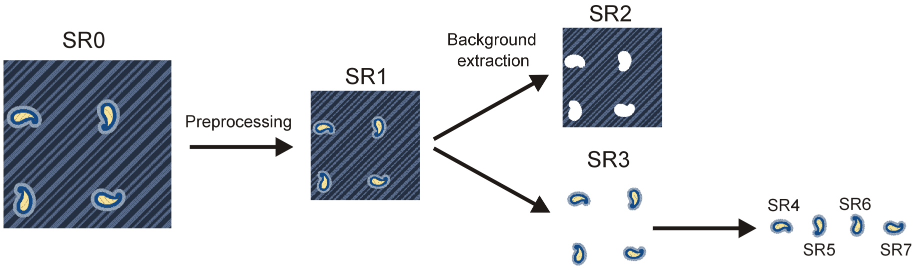


Figure 17. Example of a design segment decomposition using the *StillRegion DS* **and** *SegmentDecomposition DS*. **SR denotes a** *StillRegion*, **SD a** *SegmentDecomposition*.

Regions or elements of a design are described by means of the *Segment DS*, or more in particular for design images by the *StillRegion DS*. Hierarchical descriptions are possible: segments may be further divided into sub-segments. Decompositions are described by means of the *SegmentDecomposition DS*. An example of the use of these schemes is shown in Figure 17. The original image is denoted by SR0. After preprocessing (e.g. resizing, color quantization) we obtain a simplified image, which we denote by SR1. This region may be decomposed in a striped (background) region SR2 and a region consisting of 4 little objects (SR3). If required, SR3 may be further divided into its connected components: SR4 to SR7.

Generally, segments are allowed to overlap, and the union of the segments is not required to cover the full design. For further characterization of the organization of the segments several schemes are available to describe relations between the design regions. As such relations constitute a subset of the semantic relations, they will be discussed in the next section.

Segments may have several types of attributes, most notably the visual descriptors regarding for instance color, shape and texture of the region. The segments and their decompositions thus provide a natural framework for structuring the organization of the various types of perceptual features. In the previous example features regarding the stripes would be associated with SR3, whereas shape information can be stored as attributes of the regions corresponding to the objects. Also color information can be stored for both the entire design and the several regions separately.

Another important type of segment attribute in the context of design structure is element saliency, i.e. the extent to which an elements in a design "stands out". A mechanism to this end is provided by means of the *MatchingHint* descriptors.

4.2.3 Content semantics

Just as the content structure schemes provide a convenient framework for the low-level description of a design in terms of its regions, the content semantics description schemes provide a rich set of tools for high-level descriptions of a design in terms of its elements. The elements, the properties of the elements and the relationships between the elements are described by semantic entities, semantic attributes and semantic relations, respectively. Generally, close links exist between the regions described in the content structure schemes and the entities used in the semantic description, but the semantic content description allows various other abstract concepts to be defined as well.

Several schemes are available for the description of abstract entities of which the most relevant are the *Object*, *SemanticState* and *Concept* DSs, e.g. the *Object* DS describes perceivable semantic entities.

Semantic attributes are used to describe the semantic entities by means of labels, a textual definition, properties and for instance a link to a region in the actual design (a so-called *MediaOccurrence* attribute). Other attributes allow for the specification of the abstraction level of a given entity. This can be used for searching designs and will be discussed in more detail in the next section.

Semantic relation types are defined by means of *classification schemes* (CSs). These are provided by MPEG-7 for the definition of vocabularies for use inside descriptions. Several such schemes are already available: relations may describe for example how entities relate in a narrative (e.g. agent, patient, instrument, beneficiary), how their definitions relate to each other (e.g. generalizes, componentOf, propertyOf) or how the entities are organized with respect to spatial structure (e.g. above, left).

An example of a semantic content description is shown in Figure 18 for the decorative design of Figure 17. The decorative design is shown to consist of a background and a foreground entity. The background is associated with the abstract concept of ’striped-ness’. Two state entities further specify the quality of the stripes (their orientation, and their size). The foreground consists of a set of motifs. The type of motif set may be further clarified by for instance a variation state (indicating to what extent the motifs are varying within the set). The motifs that occur in the set, in this case so-called paisley motifs, may be described independently.

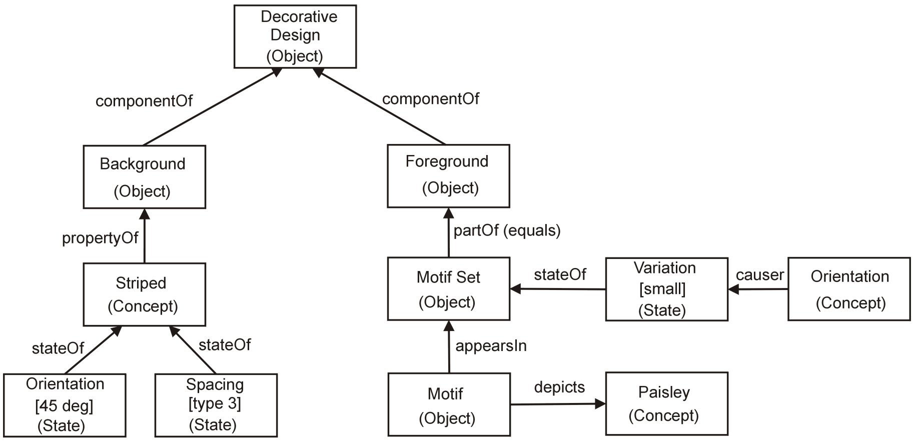


Figure 18. Example of a semantic content description of the design shown in Figure 17. The boxes represent semantical entities, the arrows semantical relations. Attributes providing links to associated segments are not shown.

Many other schemes are available which have not been mentioned yet, for instance schemes for description of design collections, user interaction or usage histories. See Benitez et al. (2002) for an overview.

4.3 High-level design description and retrieval

As shown the MPEG-7 descriptors and content structure description scheme provide a convenient framework for organizing the design metadata obtained by the various design representation methods discussed in section 2.

Ideally one would also like to be able to automatically generate high-level descriptions of the type discussed in the previous section as this level of content description is ultimately most meaningful for design retrieval. Design retrieval by semantic descriptions can then for instance take the form of listing all images that possess "at least one paisley motif", or of listing those images for which the background texture has a certain quality.

To define an efficient language for design description and query formulation the main challenge is to provide a set of well-founded constructs to describe the semantics of decorative designs. As we have already seen, entities should include (i) objects such as foreground, motif set; (ii) concepts, such as texture, variation, pattern types (e.g halfdrop, horizontal), and style types (e.g. ethnic, tropical, skin); (iii) states, e.g. variation, spacing, complexity. Basic capabilities for such description of a vocabulary of terms, the relationships that can exist between terms and the properties that such terms may have, is provided by the Resource Description Framework (RDF) Schema language (RDF (2003)). For more extensive capabilities one may also use an ontology language such as Web Ontology Language (OWL, OWL (2003))

The specification of a decorative design ontology provides the opportunity to reason about the concepts defined. Consider for example the case that Geometric, Square and Circle are classes defined in a decorative design taxonomy, and that Square and Circle are both indicated as subclasses of the Geometric class. A search based on a semantic description containing a Geometric object can then be inferred to match descriptions containing Circle or Square objects. An additional use of ontologies is the support for the representation of common knowledge. An example of such knowledge would be that tropical designs typically contain a number of bright colors.

MPEG-7 provides abstraction mechanisms that support reasoning by ontology. The most relevant is formal abstraction which describes patterns that are common to a set of examples. Formal abstractions are created by taking a specific semantic description and by replacing one or more of the entities in the description by variables using the *AbstractionLevel* attribute. In the example above we could for instance replace the paisley concept by a Curved Shape entity of *AbstractionLevel* one (level zero denotes concrete instance; levels higher than one provide abstractions of abstractions). Such description would then match any design with a certain type of striped background for which the motifs exemplify curved shapes (and for which the motifs in the set are varying by orientation).

As a final opportunity we mention the definition of a classification scheme in combination with the *Affective DS*. This provides a means to describe the esthetic appreciation of designs and allows description of measurements of affective response to variables such as balance, ratio, juxtaposition, repetition, variation, pattern, rhythm/tempo, emphasis, contrast, harmony and unity.

We conclude that MPEG-7 semantic descriptions show great promise in organizing high-level design interpretations for meaningful searching, but must note that given the current state-of-the-art fully automatic description is possible only to a limited extent, primarily as reliable recognition of meaningful objects remains a task yet to be solved.

5 Inference and learning for relevance feedback by examples

5.1 Introduction

Content-based image retrieval revolves to an important extent around the task of *interactively* and *adaptively* reaching an understanding of what the user is looking for. As discussed in the introduction, using relevance feedback may help us deal in particular with the fact that interpretation of image content is user- and task-dependent. An overview of approaches is presented in Zhou and Huang (2003); in many cases substantial gains in retrieval performances through the use of relevance feedback have been reported (e.g. Zhang and Su (2002), Cox et al. (2000), Ciocca and Schettini (1999), Rui et al. (1998) and Meilhac and Nastar (1999)).

In the following we focus on feedback in terms of example images. With this type of feedback the user is presented with a selection of images from the database; he indicates which images he considers relevant examples (positives) and which he considers to be counterexamples (negatives); next, a new selection of images based on the estimated relevance ranking is presented and the cycle may be repeated. This type of interaction is particularly natural for images: unlike for text documents, relevance of images can really be determined “at a glance”.

Many approaches to relevance feedback are based on adaptively re-weighting of feature dimensions, both for query point movement (e.g. Rocchio Jr. (1971), Rui et al. (1998)) and in similarity and relevance measures. In both cases feature variables or feature classes are assigned weights based on the feedback data. The weights should be chosen in accordance with the importance of the associated features to the user. For example, Rui et al. (1998) update weights of different feature classes by using the inverse variance of the positive examples, thereby giving higher weights to features for which the positives are close to each other. Many variants of this idea have been developed (e.g. Ciocca and Schettini (1999), Peng et al. (1999)) but generally are heuristic in nature: feature weights are assigned such that positives cluster, while negatives stay separated.

In Cox et al. (2000) a framework for Bayesian interpretation of relevance feedback data is described. At the heart of this approach lies the probabilistic modeling of the feedback data given that the user has a target image in mind and a certain selection of images is available to the user to choose from. It is assumed there that the user will pick images based on their similarity to the target, but no explicit effort is taken to find out which features are most important in the similarity measure.

Many recent approaches treat the estimation of image relevance based on the relevance feedback by examples as a machine learning or classification problem. The feedback images are taken as training samples and used to train a classifier or other learner that can be used to predict the relevance of the database images. As we have seen, typically two classes or levels of relevance are assumed. Extensions to more levels are usually straightforward (e.g. Rui et al. (1998)), but may incur the cost of a less natural interaction with the user. Examples of learning approaches to relevance feedback are: MacArthur et al. (2000, decision trees), Laaksonen et al. (2000, neural networks and self-organizing maps), Vasconcelos and Lippman (1999, Bayesian), Tong and Chang (2001, support vector machines), Wu and Manjunath (2001, nearest neighbors), Wu et al. (2000, linear discriminants) and Tieu and Viola (2004, boosting).

In the following we discuss the special structure of the relevance feedback learning problem that leads to difficulties for many of the methods mentioned earlier; we also describe a new approach which deals naturally with the fact that feedback images are typically selected based of a small set of salient properties. To be able to discuss this more clearly we first introduce the notion of *aspects*.

5.2 Aspect-based image search

As we have seen in section 2 features measure image quantities; some of these quantities will matter to image relevance and some will not (neutral features). When a feature matters we should find out which feature *values* influence relevance positively, and which negatively. Note that only for neutral features, any feature value has (approximately) the same effect on image relevance, i.e. no effect. For “relevant features”, not only will there be feature values that lead to higher perceived relevance, but there must always also be feature values that make images *less* relevant.

In our approach we will not analyze the relevance of features as a whole, but rather the relevance of an image having feature values satisfying certain conditions or belonging to a certain set. We consider for instance the influence of “high complexity”, where “high” is defined as a range of complexity feature values. We will refer to such derived binary features which model a specific perceptual quality, and which an image either has or has not, as aspects. To be more precise, we will understand an *aspect* as:

> a proposition with predicate in terms of a feature or set of features variables (which for a given image is either true false), for which we intend to resolve its effect on image relevance as a unit.

When the image features satisfy the aspect predicate we say the image *possesses*, or simply *has*, the aspect.

As mentioned, even though any constraint on feature values can be used, in practice the proposition will usually state that the image has a feature value in a certain natural range or interval. We will refer to such range or interval as the aspect *cell*. Such cells can be fixed beforehand, allowing for very fast feedback processing as we shall see later on, or be determined adaptively. The construction of aspects for different feature types is discussed in section 5.3.

We believe that when a user is searching in an image database, he does so, consciously or unconsciously, in terms of aspects. The aspects he wishes the images to possess, and which make an image at least partially relevant, we call *relevance enhancing*, or simply *relevant*, aspects. Similarly we have *neutral* and *relevance inhibiting* aspects.

As an illustrative example, suppose a user is interested in finding designs that: (i) have a blue background; (ii) have simple round motifs that are relatively far apart; and (iii) have a high contrast between motifs and ground. Depending on the available features, we can translate this to requirements in terms of aspects. Some aspects are clearly relevant, e.g. the blue-ness of the ground should be high, dominant motif shape should be round, and relative amount of background should be high. Aspects that are in opposition to the relevant aspects are relevance inhibiting, e.g. the user does not want closely spaced motifs, a ground that is red or a dominant motif shape that is square. Additional inhibiting aspects may be discovered during the feedback process, e.g. a user may decide that he is in fact not interested in yellow motifs. Other aspects are neutral as the user does not care whether images possess these. For example we may not care about the pattern in the ground: it may be plain or have some texture.

In the following we discuss the implications of feedback examples being chosen based on *partial* relevance, i.e. based solely on one or a few *salient* aspects. To this end it will be useful to quantify the saliency or importance of an aspect by measuring how often, or rather how rarely, the aspect occurs in the database. In practice we will mainly need the fraction $p_{\mathrm{db}}(a)$ of images in the database that possess a given aspect a, which in analogy to the information retrieval term of “document frequency” could also be referred to as the “image frequency” of the aspect. A natural definition of aspect saliency, in this narrow sense, can then be based directly on the definition of inverse document frequency (see for example Sparck Jones, 1972), giving the *inverse image frequency* $\mathrm{iif}(a) = \log(1/p_{\mathrm{db}}(a))$.

5.3 Aspects and feature types

More and more methods for image feature extraction become available and there is a correspondingly large heterogeneity in feature types. Several divisions of feature types can be made. Feature values may be continuous or discrete. Within the discrete, or categorical, features we have ordered (ordinal) and unordered features. An example of an ordered feature is a complexity feature divided into five levels, e.g. very simple, simple, medium, complex and very complex. An unordered feature is, for instance, "direction type" (horizontal, vertical, "diagonal grid" etc). We may also have to deal with feature spaces. As mentioned the MPEG-7 standard defines a number of visual descriptors for color, texture and local shape, where each of such descriptors consists of a feature space and a predefined similarity metric. In some cases the similarity metric consists of a complicated algorithm, for instance in the case of the MPEG-7 contour shape descriptor based on a curvature scale space approach.

In many databases manually annotated labels are available, typically of the discrete type, that have high semantic value and can be used with high reliability. Such labels may be used in hybrid systems that allow for direct selection of image subsets, but it may also be useful to let the system determine their relevance based on feedback by examples. At the other end of the spectrum we may have features that hardly have any semantic interpretation at all; such features may for instance be obtained through dimension reduction techniques such as principal component analysis or the like. Features may also be learned through interaction with the user, (see for instance Minka and Picard, 1997).

We finally observe that several features or feature spaces may be available for the characterization of more or less the same higher-order aspects, e.g. we may have several different implementations of color similarity. Which of these is most appropriate may vary and depend on the given situation.

For discrete feature types associated aspects naturally follow from the available levels. Also for single dimensional continuous features it is usually straightforward to quantize the feature, either automatically or by inspection, into a number of meaningful classes. High-dimensional feature spaces are the most interesting in this respect. Our preferred solution is to use derived features obtained by an exemplar or case-based approach. For instance, we may select a number of red example images, determine a prototype or set of prototypes (e.g. by means of LVQ, see Kohonen, 1989), and define a derived red-ness feature based on the distances to one or more prototypes. Another approach constructs data-driven aspects by mining for clusters of images in the given space. Aspects then follow from cluster membership.

Using the evidential aspect-based approach detailed below, we can treat all feature types in a unified way, and use the feedback data to establish which aspects are most promising in determining image relevance. More importantly however, it directly confronts a number of issues concerning the structure of the relevance feedback learning problem.

5.4 Special structure of the relevance feedback learning problem

The number of training samples obtained as a result of the relevance feedback process is usually small, whereas the dimension of feature space is large (typically more than 100). This makes many of the standard learning methods unsuitable for the task of relevance prediction.

Apart from this difficulty many of the learning methods do not take the special structure of the relevance feedback problem into account. We mention three issues: (i) feature value distributions are often highly skewed; (ii) the selection of examples is usually based on partial relevance; (iii) there is a lack of symmetry between positive and negative examples.

Features often have value distributions that are highly skewed. This is particularly the case for features, common for special purpose databases, measuring detailed salient properties. As examples one may think of binary features such as “contains-a-paisley”, “has-colored-stripes” or “is-a-tartan”. For many such features, the great majority of images will not possess the aspect thus leading to a highly skewed value distribution. Also, if we take a feature measuring yellow-ness, say divided into three classes: “no yellow”, “some yellow” and “very yellow”, then by far most of the database images will be in the first class, and very few will be in the last. In combination with the next issue, this skewness in the population distributions leads to a tendency for feedback data to be misleading.

When a user selects an image as feedback he generally does so based on *partial* relevance of the image. This means that he finds one or a few aspects in that image relevant; however, not all salient aspects present in the image need to be relevant, nor need all aspects of interest be present in the image. For features other than the ones by which the image was chosen, the feedback is more or less random: positive feedback is given for a certain value of the feature, where no such feedback was intended. Such examples will tend to cluster at those feature values that are most common in the database: this seems to indicate the user is interested in such values, whereas in fact he is not, thus interfering with the identification of the proper regions of relevance. See Figure 19.

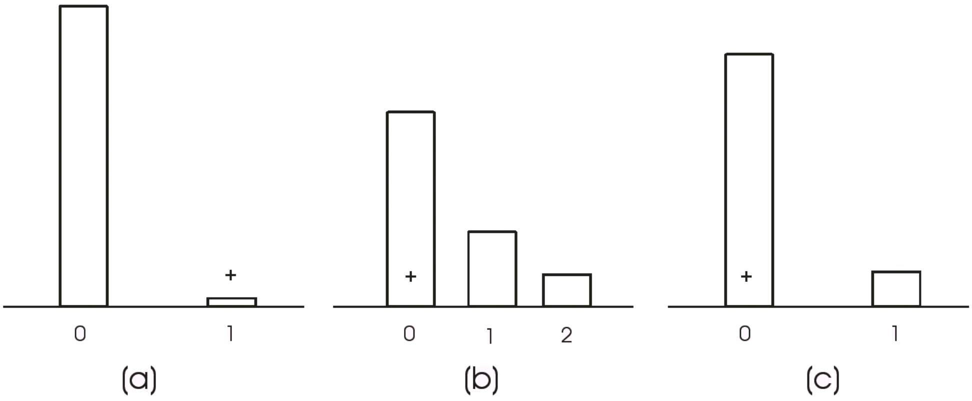


Figure 19. Bars in the diagram indicate the population probabilities of feature values in the database for three distinct features a, b and c. The plus sign indicates the feature values (for a, b,c) for an image which is selected as an example because of $a = 1$ (i.e. for having a relevant aspect $a = 1$). We suppose b is a neutral feature, and c is a feature where $c = 1$ represents another relevant aspect, which in this case does not happen to be present in the feedback example. Since the selection is based the a-value, for both cases the b and c values of this example will be (approximately) random draws from the b and c populations. Because of the skewness they seem to favor $b = 0$ and $c = 0$ consistently, which is misleading in both cases.

For negative feedback the situation is similar. The user indicates a negative example based on one or more aspects he finds undesirable. Generally we may expect he will avoid selecting images with relevant aspects to some extent, but for neutral and other relevance inhibiting aspects the feedback information is, again, often misleading.

Considering all feedback images, we can expect to encounter situations as sketched in Figure 20. Different feature types are shown (binary, ordinal, continuous); for each case we show a situation where two of the example images possess a relevant aspect in the feature under study, whereas the other four are chosen based on other relevant aspects. Note that negatives counteract the misleading clustering of positives, but most learning methods will be influenced by the unintended concentration of positives.

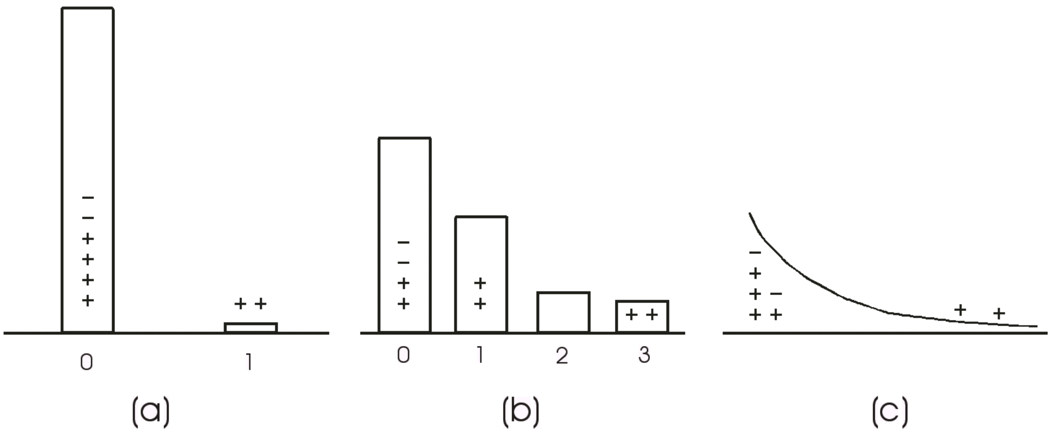


Figure 20. Diagram shows examples of the population probability (mass/density) and potential feedback for (a) a binary feature; (b) a discrete feature; and (c) a continuous feature. For each case, two positive examples are chosen based on the feature shown; the remaining examples are chosen based on other aspects.

A final issue is the lack of symmetry between positive and negative examples. Huang and Zhou (2001) state as an intuition that "the positive examples are all good in the same way, but bad examples are bad in their own ways". Though this statement may not be strictly correct in the light of the partial relevance issue discussed earlier, it is clear that the few selected negative examples are generally a very poor representation of the distribution of all negative examples. In our view no description of the class of negatives need be attempted, and examples should rather be used to purposely counteract slips of the system in seeking positive examples in misguided regions of feature space.

A useful distinction in aspects we can make in this context is that between *active* and *passive* aspects. Active aspects are aspects that the user uses explicitly in telling the system about his preferences and dislikes. What the user is looking for generally follows from a relatively small number of active enhancing aspects. Next to those there are also passive enhancing aspects, which are typically non-salient: if a user is looking for very yellow images then a "little-or-no-red" aspect would be enhancing but only in a passive sense. As follows from the lack of symmetry in positive and negative aspects, the number of inhibiting (salient) aspects is usually larger than the number of salient enhancing aspects. However, most inhibiting aspects tend to be passive. Which of the inhibiting aspects become active is not only be determined by explicit wishes of the user but also to some extent by chance, e.g. a group of red images shows up in the selection, at which the user decides to indicate he is not interested in this color.

In the following we will describe our approach to the interpretation of feedback data. We will use the feedback data first and foremost to establish which aspects matter most to the perceived relevance. For each available aspect we determine the likelihood ratio of the hypothesis that the aspect enhances relevance, relative to the hypothesis that relevance is independent of the aspect. Taking this approach has the benefit that relevance assignment is based not only on clustering behavior of positives and negatives, but is also compared to clustering behavior of random database images. This leads to a natural emphasis on salient aspects thereby solving the problems of partial relevance discussed earlier.

In addition, by taking into account the population distribution, we are not dependent on negative examples to down-weight positives that cluster at aspects with low saliency. This means negatives can be used to indicate which aspects are not desired, but are not required for the sole purpose of getting sufficient data for classification.

Finally, the use of the likelihood ratio for the evaluation of relative evidential support has a strong foundation in statistics, and allows for detailed modeling of the user's behavior in providing feedback. This, in turn, allows for inference solutions that are tailored to the needs of particular interaction schemes.

5.5 Measuring evidential support by likelihood ratios

In the following we are interested in analyzing the evidential support offered by data in comparing hypotheses. Rather than stating the problem in terms of the standard Neyman-Pearson approach in deciding between hypotheses using the likelihood ratio as a test statistic, we here emphasize a direct evidential interpretation of the likelihood ratio.

Royall (1997) and Royall (2000) make a strong case for the view that evidential support should not be measured for a hypothesis in isolation, but should preferably be considered relative to other hypotheses. The strength of such relative evidential support is quantified by the likelihood ratio. Hacking(1965) states this so-called **Law of Likelihood** as follows:

> If hypothesis A implies that the probability that a random variable X takes the value x is $p_A(x)$, while hypothesis B implies that the probability is $p_B(x)$, then the observation $X = x$ is evidence supporting A over B if only if $p_A(x) > p_B(x)$, and the likelihood ratio, $p_A(x)/p_B(x)$, measures the strength of that evidence.

If we have a parameterized probability model for X with distributions indexed by a parameter θ, then an observation $X = x$ generates a likelihood function $L(\theta)$. The law of likelihood then explains how to use this function: for any two parameter values θ_1 and θ_2, the ratio $L(\theta_1)/L(\theta_2)$ measures the strength of evidence, $X = x$ in support of θ_1 vis-á-vis θ_2.

In this article we propose to use feedback data (the positive and negative example images) in this way by comparing a number of hypotheses on the relation between a given aspect and image relevance. These hypotheses basically state either that an aspect is independent of image relevance (i.e. the aspect is neutral), or that the aspect is relevance enhancing or inhibiting in a certain way. Each of the hypotheses leads to a likelihood value for the feedback data. The law of likelihood quantifies the relative evidential support for any *pair* of hypotheses; in particular we will be interested if the maximum likelihood hypothesis stands out sufficiently from the alternative hypotheses.

An important question is how to interpret the likelihood ratio values. For instance, which ratios can be said to provide only weak evidence, and which strong evidence. As discussed in Royall (1997) there are various ways to develop a quantitative understanding of likelihood ratios.

The first is to compare ratios to ones obtained in canonical experiments where intuition is strong. As an example of such an experiment, suppose we have two identical urns, one containing only white balls, the other containing equal numbers of black and white balls. One urn is chosen and we draw a succession of balls from it, after each draw returning the ball to the urn and thoroughly mixing its contents. We then have two hypotheses about the contents of the urn, and the observations are the evidence.

Suppose we draw 3 balls in succession which are all white, then the likelihood ratio is $2^3 = 8$ favoring the all-white-ball hypothesis over the mixed-balls hypothesis. Similarly, if we draw n balls in succession which are all white, the likelihood ratio is 2^n. Of course there is no sharp transition between weak and strong evidence, but one may use such experiments to agree on certain benchmarks: Royall (1997) proposes to use a likelihood ratio of 8 as representing “fairly strong” evidence and 32 as “strong” evidence favoring one hypothesis over the other. Note that a likelihood ratio of 32 corresponds to drawing 5 successive white balls in the canonical experiment. Similarly likelihood ratios around 1 represent “weak” evidence.

A second way of understanding likelihood ratios is by considering their effect in transforming prior into posterior odds ratios. We have

$$\frac{\Pr(A|X=x)}{\Pr(B|X=x)} = \frac{p_A(x)}{p_B(x)} \frac{\Pr(A)}{\Pr(B)}, \tag{8}$$

where $\Pr(A)$ is the prior probability that hypothesis A is true; $\Pr(A|X=x)$ the posterior, and $p_A(x)$ is the likelihood of the data x under hypothesis A.

So for each prior probability ratio $\frac{\Pr(A)}{\Pr(B)}$, the likelihood ratio $\frac{p_A(x)}{p_B(x)}$ tells us how the probability ratio changes after observing the data. For instance a likelihood ratio of 4 always produces a fourfold increase in the probability ratio.

Finally we mention a re-interpretation of likelihood values based on the observation that evidence may be *misleading*, i.e. it may happen that data represents strong evidence in favor of one hypothesis whereas, in fact, another hypothesis is true. Generally this cannot occur very often as is shown in Royall (2000) and below we will compute exact probabilities for events such as obtaining strong evidence for neutral aspects being enhancing. The expressions discussed there have strong links to standard levels of significance and hypothesis test power.

To summarize, we compare hypotheses based on their likelihood values; in the case of analyzing the partial relevance of aspects given the feedback data we compare hypotheses that the aspect is enhancing, neutral or inhibiting. Ideally we will be able to accurately model the probability of a stochastic variable representing the feedback data for each of the hypotheses, such that its measurement will provide strong evidence for one of the hypotheses most of the time. The design of such variables and models is not trivial, and will be taken up in the next section.

5.6 An evidential approach to relevance feedback by examples

As explained, the basic setup in relevance feedback is such that a user selects images from the database to indicate his preferences and dislikes. Selection is facilitated by presenting the images in clickable selection screens each consisting of a grid of a fixed number of, say 30, thumbnail images. The number of images inspected may be larger as the user can leaf through the selection screens. Also additional selection screens may be available, for instance offering ‘informative images’, see section 5.6.4. The sequential ordering of the images is either random in the first cycle, or based on the relevance ranking in the subsequent cycles. The positive examples and (negative) counterexamples are collected in the collection box, consisting of the positive and negative *feedback image sets*.

At each cycle of the feedback process the user updates the examples in the feedback image sets by either: (i) selecting new images as positive or negative examples adding them to their respective sets; (ii) by removing images from the feedback image sets, i.e. the sets are preserved unless specific images are no longer deemed representative enough and are deleted explicitly.

As the user selects the example images based on partial relevance, i.e. based on one or a few enhancing or inhibiting aspects, it is natural to use the feedback data foremost to establish the relevance effect of the various aspects (i.e. as either enhancing, inhibiting or neutral). At the end of each cycle we thus analyze the relation between aspects and relevance by using models for the three types of effects.

The construction of the models will be based mainly on the following idea: as the user selects an image as feedback example based on one or a few enhancing or inhibiting aspects, possession of the remaining aspects will approximately follow the distribution (of aspect possession) in the database. Corresponding to the models, we formulate the following hypotheses to explain the occurrence of each of the aspects in the feedback images:

1. the *independence*-hypothesis, or H_0-hypothesis: whether the image has the aspect is independent of whether the image is relevant;

2. the *relevance enhancing*-hypotheses, denoted by H_K: possession of the aspect enhances image relevance;

3. the *relevance inhibiting*-hypotheses, denoted by H_{-K}: possession of the aspect inhibits image relevance.

K denotes the number of images in the positive (enhancing case) or negative feedback image set (inhibiting case) for which the user's selection was directly influenced by the aspect under study. This means we will consider a sequence of hypotheses $\ldots, H_{-1}, H_0, H_1, H_2, \ldots$ where the actual number of hypotheses to be analyzed depends on the number of images in the feedback image set actually possessing the aspect under study. This will be explained in more detail below by discussing the models for each of the effects in turn.

5.6.1 Modeling feedback under the independence hypothesis

The H_0-hypothesis states that the relevance of the image and the aspect predicate are *independent*. In this case all feedback images have been chosen based on aspects other than the aspect under consideration, and this means that for all feedback images possession of this aspect will be distributed approximately as for a random image from the database. We thus model feedback image possession of the aspect as a Bernoulli variable with probability p_{db}, the fraction of images in the database which have the aspect.

In the following we will assume the number of positive and negative images selected to be given, and consider for each of these images if they possess the aspect or not, i.e. whether or not the associated feature values of the example images satisfy the aspect condition.

Let n^+ (n^-) be the total number of positive (negative) images selected, and N^+ (N^-) be the number of positives (negatives) that possess the aspect.

We then model both the number of positives and negatives occurring in the cell as binomial variables with probability parameter p_{db}:

$$N^+ \sim B(n^+, p_{\mathrm{db}}), \qquad \text{and} \qquad N^- \sim B(n^-, p_{\mathrm{db}}). \tag{9}$$

To see this, imagine a large urn containing a ball for each image of the database. White balls represent images which have the aspect, and black balls images which don't; the urn will then have a fraction p_{db} of white balls. If we now draw n^+ balls, the number of balls N^+ that is white is modeled well by the binomial distribution $B(n^+, p_{\mathrm{db}})$.

The total probability mass function $p_0(x)$ for the feedback data $x = (N^+, N^-)$ is the product of the probabilities for N^+ and N^-.

5.6.2 Modeling feedback under partial relevance: enhancing and inhibiting hypotheses

Relevance enhancing hypotheses

The total numbers of positives (n^+), negatives (n^-) are again taken to be fixed. Given that an aspect is relevant, we expect that a few, say $\tilde{N}^+$, of the n^+ selected positive images have been chosen based to some extent on this aspect. As the remaining positives are selected because of other relevant aspects, their aspect possession will then be distributed similar as under the independence hypothesis.

The H_K hypothesis now states that the aspect is relevance enhancing, and that $\tilde{N}^+ = K$ of the feedback images have been selected, at least partially, based on this aspect. The $(n^+ - \tilde{N}^+)$ remaining positives are chosen independently from the aspect, so we have

$$(N^+ - K) \sim B(n^+ - K, p_{\text{db}}), \tag{10}$$

or

$$p(N^+|\tilde{N}^+ = K) = \left(\begin{array}{c} n^+ - K \\ N^+ - K \end{array} \right) p_{\text{db}}^{(N^+ - K)} (1 - p_{\text{db}})^{(n^+ - N^+)}, \tag{11}$$

for $N^+ >= K$, and $p(N^+|\tilde{N}^+ = K) = 0$ for $N^+ < K$.

To obtain a model for the probability distribution of N^- we assume that negative examples have a probability to possess the aspect as under the independence hypothesis, i.e.

$$N^- \sim B(n^-, p_{\text{db}}). \tag{12}$$

The total probability mass function $p_K(x)$ for the feedback data $x = (N^+, N^-)$ under hypothesis H_K is again the product of the probabilities for N^+ and N^-.

Relevance inhibiting hypotheses

The model for relevance inhibiting hypotheses is derived analogously to the enhancing case. For H_{-K} we assume that $\tilde{N}^- = K$ of the negative images were chosen based on the aspect, giving that

$$(N^- - K) \sim B(n^- - K, p_{\text{db}}). \tag{13}$$

Assuming aspect possession for the positives as under independence leads to $p_{-K}(x)$.

5.6.3 Estimation of relevance

Aspect selection

Before we can estimate relevance of images in the database based on the feedback data obtained, we must first decide which aspects to take into account.

In our approach we take only those aspects for which either a relevance enhancing or a relevance inhibiting aspect is sufficiently well supported in comparison to the independence hypothesis. Here evidential support will be taken in the sense of section 5.5, i.e. measured by means of the likelihood ratio of the respective hypotheses.

Let $p_0(x)$ be the likelihood of the feedback data under the independence hypothesis, and $p^+(x)$ and $p^-(x)$ the maximum likelihood values of the data under the enhancing and inhibiting hypotheses respectively, i.e.

$$p^+(x) = \max_{K>0} p_K(x), \quad \text{and,} \quad p^-(x) = \max_{K<0} p_K(x). \tag{14}$$

We take T to be our main decision threshold variable. If either $p^+(x)/p_0(x) >= T$ or $p^-(x)/p_0(x) >= T$ we accept the aspect as enhancing or inhibiting, respectively, i.e. we select such aspects to be taken into account in the relevance estimation. Note that the first likelihood ratio basically measures if the number of positives with the aspect is unexpectedly high, and the second ratio whether the number of negatives with the aspect is unexpectedly high. It may sometimes happen that both are the case; this special case will be further discussed below. Also note that the enhancing and inhibiting hypotheses model *active* aspects. Passive enhancing or inhibiting aspects will behave as neutral aspects and will thus, as intended, not be selected.

The threshold T can be directly interpreted in the sense described in section 5.5. For instance if we have an aspect with image frequency $p_{\mathrm{db}} = 0.5$ and take $T = 8$, one way to get the aspect to be accepted would be to select 3 out of 3 images with this aspect, i.e. by taking three positive examples each having the aspect. For more salient aspects fewer positives and a lower fraction of aspects having the aspect are required.

A more quantitative approach is to analyze probabilities of obtaining misleading evidence. In the following we discuss the comparison between neutral and enhancing aspects, but the same results apply to the neutral-inhibiting comparison.

We can compute the probability of accepting an aspect as enhancing, when the aspect is, in fact, neutral. To this end, we first note that for given (n^+, n^-) and p_{db}, each combination $X = (N^+, N^-)$ leads to a fixed likelihood ratio

$$\mathrm{LR}^+(X) = \frac{p^+(X)}{p_0(X)}. \tag{15}$$

Now assuming that an aspect is neutral means that we know the distribution of $X = (N^+, N^-)$, as $X \sim p_0$. The probability of obtaining misleading evidence indicating the aspect is enhancing for a threshold T is thus given by $\mathrm{Pr}(\mathrm{LR}(X) >= T)$ given that X has distribution p_0. As an example, for an aspect with image frequency 0.1 (i.e. representing a quality that occurs once every 10 images), with $n^+ = 7$ we find that the probability of obtaining this type of misleading evidence is equal to 0.026 (for $T = 8$ and $T = 16$) and equal to 0.0027 for $T = 32$. This is, of course, a general pattern: when we increase T the probability of mistakenly deciding an aspect is enhancing decreases. However, there is a trade-off as increasing T also leads to an increased probability of obtaining evidence not sufficiently supporting an enhancing aspect to be enhancing. For instance, it turns out that the probability of mistakenly treating an enhancing aspect in this setup as neutral is zero for $T = 8$, or $T = 16$, but is equal to a large 0.64 for $T = 32$.

It is interesting to note that the Neyman-Pearson theorem (Neyman and Pearson, 1933) guarantees that for a given probability of falsely rejecting the neutral hypothesis, the decision based on the likelihood ratio has optimal power in deciding between the neutral hypothesis and any of the individual alternative hypotheses.

Further analysis has shown that:

- Very non-salient aspects (with $p_{\mathrm{db}} > 0.5$ say) are generally hard to discern from neutral, i.e. there is a large probability that evidence will be found for these aspects to be neutral even when they are not. These aspects might thus as well be ignored.
- $T = 8$ is a good general purpose value for the threshold.
- For very salient aspects (with $p_{\mathrm{db}} < 0.01$ say) it is possible to raise the thresholds, thereby reducing chances of mistaking neutral aspects for enhancing aspects, without running the risk of missing relevant aspects.

Of course we could also optimize the threshold for every given configuration in terms of (n^+, n^-), and p_{db}. Note however, that whether an aspect is enhancing does not correspond to a unique hypothesis. We thus need to make an additional assumption on which K-value or combination of K-values to base our distribution of X. For the experiment described above we have used the assumption that the X is distributed according to a K-value that is 2 higher than the expected number under the independence hypothesis. Using simulation studies we intend to further analyze this issue and also explore the effects of the aspect correlations mentioned below.

As a final issue it was mentioned that it may happen that both the enhancing and inhibiting hypotheses receive strong support relative to the independence hypothesis, giving a so-called *entangled* aspect. Two conflicting explanations for this state of affairs are possible: (i) the user has strongly emphasized an aspect with his positive images without intending to do so (the aspect is neutral or inhibiting); as a reaction he has chosen an approximately equal fraction of negatives to undo this; (ii) the user actually finds the aspect relevant, but has nevertheless selected a number of negative images with this aspect (for example because he had no other choice). Note that this issue is not unique to our approach, but is a general problem in relevance feedback analysis. In our implementation we currently solve the problem using two strategies. The first is to demand not only sufficient support relative to the independence hypothesis but also relative to the opposing hypothesis. The second strategy is to use the direct feedback approach described in the introduction. This consists in presenting the user, at his request, with an overview of selected enhancing, inhibiting and entangled aspects. In this overview the user may confirm or reject the presented results. For each aspect he may indicate whether he considers it enhancing, inhibiting, or neutral. For enhancing and inhibiting aspects two levels of acceptance can be indicated: either selection in an AND-sense, or selection in an OR-sense. When an aspect is selected in an AND-sense this means that from then on only images will be considered (and shown) that possess that particular aspect, and images without it are no longer taken into account. The OR-sense means that the aspect is treated as any other aspect that is selected based on the statistical analysis: it will provide a contribution to the relevance as will be discussed below, but is not strictly required for an image to be considered relevant.

Relevance estimation

For fixed aspect cells we use a matrix M with columns of boolean variables to indicate whether images have a given aspect or or not. From M we compute $p_{\rm db}$ for each aspect as the ratio of ones in each column.

We can determine N_j^+ and N_j^- from the image index sets S^+ and S^- of positive and negative examples, using sums $\sum_{i=1}^{n^+} M(S_i^+, j)$ and $\sum_{i=1}^{n^-} M(S_i^-, j)$ respectively. Computing likelihood ratios LR_j^+ and LR_j^- for each aspect is a simple matter of substituting the values for N_j^+, N_j^- in the formulas for $p^+(x)$, $p^-(x)$ and $p_0(x)$. Also note that analyzing a number of enhancing and inhibiting hypotheses does not lead to any substantial additional computational expense, as we can compute associated likelihood values sequentially (for instance for positive K) by

$$p_K(x) = \frac{(N^+ - K + 1)}{(n^+ - K + 1)} \frac{p_{K-1}(x)}{p_{\rm db}}, \quad 1 \leq K \leq N^+. \tag{16}$$

To obtain a prediction of image relevance to get a new selection of most relevant images we consider only the aspects that receive strongest support. Motivated by the results discussed above, we demand a likelihood ratio greater than 8. Let A^+ be the index set of accepted enhancing aspects, and A^- be the index set of accepted inhibiting aspects, then the predicted image relevance rel_i for image i is given by $\mathrm{rel}_i = \sum_j M(i, A_j^+) - \sum_j M(i, A_j^-)$.

Note that, of course, the decision of taking into account an aspect need not so black-and-white; for instance we may down-weight the effect of aspects that show promise but for which the evidence is just not strong enough. On the other hand, one should take into consideration that the strength of evidence cannot be used directly as weighting factor in relevance prediction as for aspects receiving strong evidence for their relevance their precise strength of evidence no longer matters; in other words, weighting factors should saturate for high evidence ratios. A saturating weighting function that could be used to this end is for instance $w(\mathrm{LR}) = 1 - \exp(-\alpha \mathrm{LR})$, where α determines the saturation rate.

Perhaps even more importantly, however, given that typically we have a very large number of aspects, mixing the effect of many uncertain aspects may drastically deteriorate performance as the many aspect that are of no particular relevance will dominate the effect of the few truly relevant aspects. We thus choose to include only those aspects for which the evidence is sufficiently strong. Also various types of corrections for correlations between aspects are possible, (e.g. by aspect grouping, or shared scoring) these are, however, beyond the scope of this chapter.

5.6.4 Selection of informative images

An infamous issue in content-based image retrieval is the "page-zero problem": a user has a certain image or class of images in mind, but due to the relative small fraction of the database shown he cannot find images to get him started. We can aid in this situation by showing the user not only a selection of images based on ranking by relevance, but additionally provide images with aspects the user has not yet seen up to that point. Such images are also informative from the inference perspective as the system cannot determine the relevance of aspects for which the user has had no chance to provide feedback.

Let S be the index set of n^S images shown up until that point from which the user has been allowed to make his selection, of which N^S images have the aspect.

To construct an informative selection, we sort the aspects based on their associated N^S value, i.e. the number of images shown to the user that possess the aspect. From this ranking we determine a set $\tilde{A}$ of most informative aspects.

Next, we rank the images that satisfy at least one of these aspects by increasing value of the total number of such aspects they have. This allows the user to provide partial relevance feedback on informative aspects, while minimizing the risk that he provides misleading feedback on other aspects.

6 Conclusion and outlook

With the current state-of-the-art in image content description and feature extraction, meaningful retrieval in specialized image databases still depends to an important extent on explicit modeling of domain knowledge. For many low-level perceptual characteristics, e.g. with respect to color and texture, standard solutions are available. However, as the user generally thinks in terms of high level concepts, such characterizations are generally insufficient in satisfying his wishes.

Features are required that characterize the structure of the image in terms of properties and relations between the elements in the images. For instance, we have seen that for decoration designs, users may wish to find designs with certain specific types of objects, of a certain size etc. Modeling of domain knowledge should be done as generically as possible in order to avoid re-inventing the wheel for every single aspect that may play a role in the user's perception. Research is thus needed in developing flexible representations that may be re-used for various characterization tasks. In our view, a key requirement to this end is a robust system of object detection and recognition. By exploiting domain knowledge it is sometimes possible to achieve a satisfactory performance in this respect as we have seen in the example of the figure-ground segregation approach. However, further work remains to be done in particular in coping with the intricate interactions of various grouping principles and in being able to deal with the most unambiguous clues first.

Given the flexible representations, machine learning techniques can establish the precise relations required for the characterization task at hand.

Ontologies facilitate standardization by providing a vocabulary of shared terms together with a structure of feasible properties and relations by which images in a domain are described. The development of such description ontologies is also a very useful exercise in gaining insight into which aspects need to be captured automatically.

Once images can be described in terms of higher level aspects, we must still face the fact that which of these aspects matters most is generally user- and task-dependent. Using relevance feedback techniques this issue may be resolved through natural interaction with the system. To this end we described a statistically principled approach, which is directly aimed at assessing which image aspects determine relevance, and which takes into account the special structure of feedback data.

Acknowledgments

This work has been partially funded by the European Commission under the IST Programme of the Fifth Framework (Project nr. IST-2000-28427). We thank Sophis Systems NV and Pianezza Srl for supplying the decoration design test collections.

References

Benitez, A. B., J. M. Martinez, H. Rising, and P. Salembier (2002). Description of a single multimedia document. In B. Manjunath, P. Salembier, and T. Sikora (Eds.), *Introduction to MPEG-7 – multimedia content description interface*, pp. 111–138. Chichester, England: John Wiley and Sons, Ltd.

Bigün, J. and G. Granlund (1987). Optimal orientation detection of linear symmetry. *Proc. 1st Int. Conf. Computer Vision*, 433–438.

Bober, M., F. Preteux, and W.-Y. Y. Kim (2002). Shape descriptors. In B. Manjunath, P. Salembier, and T. Sikora (Eds.), *Introduction to MPEG-7 – multimedia content description interface*, pp. 231–260. Chichester, England: John Wiley and Sons, Ltd.

Boggess, A. and F. J. Narcowich (2001). *A first course in wavelets with Fourier analysis*. Upper Saddle River, New Jersey 07458: Prentice-Hall Inc.

Bovik, A., M. Clark, and W. Geisler (1990). Multichannel texture analysis using localized spatial filters. *IEEE Trans. on Pattern Analysis and Machine Intelligence 12*(12), 55–73.

Choi, Y., C. S. Won, Y. M. Ro, and B. Manjunath (2002). Texture descriptors. In B. Manjunath, P. Salembier, and T. Sikora (Eds.), *Introduction to MPEG-7 – multimedia content description interface*, pp. 213–230. Chichester, England: John Wiley and Sons, Ltd.

Ciocca, G. and R. Schettini (1999). Using a relevance feedback mechanism to improve content-based image retrieval. In D. Huijsmans and A. Smeulders (Eds.), *Visual information and information systems*, pp. 107–114. Springer Verlag.

Cox, I., M. Miller, T. Minka, and T. Papathomas (2000). The Bayesian image retrieval system, PicHunter: Theory, implementation, and psychophysical experiments. *IEEE Trans. Image Processing 9*(1), 20–37.

Duda, R. and P. Hart (1973). *Pattern classification and scene analysis*. New York, USA: John Wiley and Sons, Inc.

Eakins, J., J. Boardman, and M. Graham (1998). Trademark image retrieval by shape similarity. *IEEE Multimedia 5*(2), 53–63.

Flickner, M., H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker (1995). Query by image and video content: The QBIC System. *IEEE Computer 28*(9), 23–32.

Freeman, W. and E. Adelson (1991). The design and use of steerable filters. *IEEE Trans. on Pattern Analysis and Machine Intelligence 13*(9), 891–906.

Freixenet, J., X. Munoz, D. Raba, J. Marti, and X. Cufi (2002). Yet another survey on image segmentation: Region and boundary information integration. *Proceedings ECCV 2002 III,* 408–422.

Gimel'farb, G. and A. Jain (1996). On retrieving textured images from an image database. *Pattern recognition 29*(9), 1461–1483.

Goldfarb, P. and P. Prescod (2001). *The XML Handbook (3rd edition)*. NJ: Prentice Hall.

Gotlieb, C. and H. Kreyszig (1990). Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics, and Image Processing 51*(1), 70–86.

Huang, T. and S. Zhou (2001, October). Image retrieval with relevance feedback: From heuristic weight adjustment to optimal learning methods. *Proc. IEEE Int. Conf. on Image Processing (ICIP)*.

Huiskes, M. and E. Pauwels (2003). Segmentation by color coalition labeling for figure-ground segregation in decoration designs. *Proceedings of the Third International Symposium on Image and Signal Processing and Analysis (ISPA),* 84–90.

Huiskes, M., E. Pauwels, P. Bernard, H. Derumeaux, P. Vandenborre, L. Van Langenhove, and S. Sette (2003, June). Metadata for decorative designs: application of mpeg-7 in automatic design interpretation. *Proceedings of the World Textile Conference and 3rd Autex Conference*, 502–506.

Kass, M. and A. Witkin (1987). Analyzing oriented patterns. In M. Fischler and M. Kaufman (Eds.), *Readings in Computer Vision*, pp. 268–276.

Kohonen, T. (1989). *Self-organization and associative memory (3rd edition)*. Berlin: Springer-Verlag.

Laaksonen, J., M. Koskela, and E. Oja (2000, December). PicSOM: Self-organizing maps for content-based image retrieval. *Pattern Recognition Letters 21*(13/14), 1199–1207.

Laine, A. and J. Fan (1993). Texture classification by wavelet packet signature. *IEEE Trans. on Pattern Analysis and Machine Intelligence 15*(11), 1186–1191.

Lin, H., L. Wang, and S. Yang (1997a). Color image retrieval based on hidden Markov models. *IEEE Trans. Image Processing 6*(2), 332–339.

Lin, H., L. Wang, and S. Yang (1997b). Extracting periodicity of a regular texture based on autocorrelation functions. *Pattern Recognition Letters 18*, 433–443.

Liu, F. and R. Picard (1996). Periodicity, directionality, and randomness: Wold features for image modeling and retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence 18*(7), 517–549.

Liu, Y., R. Collins, and Y. Tsin (2004). A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Trans. Pattern Analysis and Machine Intelligence 26*(3), 354–371.

MacArthur, S., C. Brodley, and C. Shyu (2000). Relevance feedback decision trees in content-based image retrieval. *IEEE Workshop on content-based access of image and video libraries*, 68–72.

Manjunath, B. and W. Ma (1996). Texture features for browsing and retrieval of image data. *IEEE Trans. on Pattern Analysis and Machine Intelligence 18*(8), 837–842.

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

Meilhac, C. and C. Nastar (1999). Relevance feedback and category search in image databases. *Proc. Int'l Conf. Multimedia Computing and Systems*, 512–517.

Minka, T. and R. Picard (1997). Interactive learning using a "society of models". *Pattern recognition 30*(4), 565–581.

Mojsilovic, A., J. Hu, and E. Soljanin (2002, November). Extraction of perceptually important colors and similarity measurement for image matching, retrieval, and analysis. *IEEE Trans. on Image Processing 11*(11), 1238–1248.

MPEG7 (2003). MPEG7: Multimedia Content Description Interface. `http://www.chiariglione.org/mpeg/`.

Neyman, J. and E. Pearson (1933). On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society, Series A 231*, 289–337.

Ohm, J.-R., L. Cieplinski, H. J. Kim, S. Krishnamachari, B. Manjunath, D. S. Messing, and A. Yamada (2002). Color descriptors. In B. Manjunath, P. Salembier, and T. Sikora (Eds.), *Introduction to MPEG-7 – multimedia content description interface*, pp. 187–212. Chichester, England: John Wiley and Sons, Ltd.

OWL (2003). Web Ontology language. `http://www.w3.org/2001/sw/WebOnt/`.

Pal, N. and S. Pal (1993). A review on image segmentation techniques. *Pattern Recognition 26*(9), 1277–1294.

Pass, G., R. Zabih, and J. Miller (1996, November). Comparing images using color coherence vectors. *Fourth ACM Conference on Multimedia*, 65–73.

Pauwels, E., M. Huiskes, P. Bernard, K. Noonan, P. Vandenborre, P. Pianezza, and M. De Maddelena (2003). FOUNDIT: Searching for decoration designs in digital catalogues. *Proceedings of the 4th European Workshop on Image Analysis for Multimedia Interactive Services*, 541–544.

Peng, J., B. Bhanu, and S. Qing (1999). Probabilistic feature relevance learning for content-based image retrieval. *Computer Vision and Image Understanding 75*(1/2), 150–164.

Pratt, W. (1991). *Digital Image Processing (2nd edition)*. John Wiley and Sons.

Randen, T. and J. Husoy (1999, April). Filtering for texture classification: a comparative study. *IEEE Trans. Pattern Analysis and Machine Intelligence 21*(4), 291–310.

Ranguelova, E., M. Huiskes, and E. Pauwels (2004). Towards computer-assisted photo-identification of humpback whales. *Proceedings of ICIP 2004*.

RDF (2003). Resource Description Frameworks. `http://www.w3.org/RDF/`.

Reed, T. and J. Du Buf (1993, May). A review of recent texture segmentation and feature extraction techniques. *Source CVGIP: Image Understanding archive 57*(3), 359–372.

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

Royall, R. (1997). *Statistical evidence: a likelihood paradigm*. Monographs on statistics and probability. London: Chapman and Hall.

Royall, R. (2000). On the probability of obserserving misleading statistical evidence. *Journal of the American Statistical Association 95*(451), 760–780.

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

Russ, J. (1995). *The image processing handbook (2nd edition)*. CRC Press.

Salembier, P. and J. R. Smith (2002). Overview of multimedia description schemes and schema tools. In B. Manjunath, P. Salembier, and T. Sikora (Eds.), *Introduction to MPEG-7 – multimedia content description interface*, pp. 83–94. Chichester, UK: John Wiley and Sons, Ltd.

Schattschneider, D. (1978). The plane symmetry groups: their recognition and notation. *American Mathematical Monthly 85*, 439–450.

Smeulders, A., M. Worring, S. Santini, and R. Jain (2000). Content-based image retrieval at the end of the early years. *IEEE Trans. on Pattern Analysis and Machine Intelligence 22*(12), 20–37.

Soille, P. (1999). *Morphological Image Analysis*. Berlin, Germany: Springer.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28*(1), 11–21.

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

Tieu, K. and P. Viola (2004). Boosting image retrieval. *International Journal of Computer Vision 56*(1/2), 17–36.

Tong, S. and E. Chang (2001). Support vector machine active learning for image retrieval. *Proc. of 9th ACM Int'l Conf. on Multimedia*, 107–118.

Van Droogenbroeck, M. and H. Talbot (1996). Fast computation of morphological operations with arbitrary structuring elements. *Pattern Recognition Letters 17*(14), 1451–1460.

Vasconcelos, N. and A. Lippman (1999). Learning from user feedback in image retrieval. *NIPS 99*, 977–986.

Veltkamp, R. and M. Hagedoorn (2001). State-of-the-art in shape matching. In M. Lew (Ed.), *Principles of Visual Information Retrieval*, pp. 87–119. Springer.

Wertheimer, M. (1923). Untersuchungen zur Lehre von der Gestalt. II. *Psychologische Forschung 4*, 301–350.

Wu, P. and B. Manjunath (2001). Adaptive nearest neighbor search for relevance feedback in large image databases. *Proc. of 9th ACM Int'l Conf. on Multimedia*, 89–97.

Wu, Y., Q. Tian, and T. Huang (2000). Discriminant EM algorithm with application to image retrieval. *IEEE CVPR*, 1222–1227.

Zhang, H. and Z. Su (2002). Relevance feedback in CBIR. In X. Zhou and P. Pu (Eds.), *Visual and Multimedia Information Systems*, pp. 21–35. Kluwer Academic Publishers.

Zhou, X. and T. Huang (2003, April). Relevance feedback in image retrieval: a comprehensive review. *ACM Multimedia Systems Journal 8*(6), 536–544.