Low-dimensional and comprehensive color texture description

Susana Alvarez[a,*], Anna Salvatella[c], Maria Vanrell[b,c], Xavier Otazu[b,c]

[a]*Rovira i Virgili University, Department of Computer Science & Mathematics, Campus Sescelades, Avinguda dels Països Catalans, 26, 43007 Tarragona, Spain*
[b]*Dept. Ciències de la Computació - Universitat Autònoma de Barcelona*
[c]*Computer Vision Center, Edifici O, Campus UAB, 08193 Bellaterra, Barcelona, Spain*

Abstract

Image retrieval can be dealt by combining standard descriptors, such as those of MPEG-7, which are defined independently for each visual cue (e.g. SCD or CLD for Color, HTD for texture or EHD for edges). A common problem is to combine similarities coming from descriptors representing different concepts in different spaces. In this paper we propose a color texture description that bypasses this problem from its inherent definition. It is based on a low dimensional space with 6 perceptual axes. Texture is described in a 3D space derived from a direct implementation of the original Julesz's Texton theory and color is described in a 3D perceptual space. This early fusion through the blob concept in these two bounded spaces avoids the problem and allows us to derive a sparse color-texture descriptor that achieves similar performance compared to MPEG-7 in image retrieval. Moreover, our descriptor presents comprehensive qualities since it can also be applied either in segmentation or browsing: (a) a dense image representation is defined from the descriptor showing a reasonable performance in locating texture patterns included in complex images; (b) a vocabulary of basic terms is derived to build an intermediate level descriptor in natural language improving browsing by bridging semantic gap.

Keywords: color texture descriptors, basic terms vocabulary, retrieval, segmentation, browsing

1. Introduction

Due to the growth in size of image collections and the need to retrieve semantically-relevant images from them, the development of effective systems for image retrieval has acquired great importance since the early 90s. Since then, the studies on the development of content-based image retrieval systems have widely increased. The goal of these content-based image retrieval (CBIR) systems is to represent and to index image databases using the visual content of the images such as color, shape, texture and spatial layout, so low-level image feature extraction is the basis of CBIR systems. Usually multidimensional feature vectors are used to describe these contents. The descriptors can either be extracted from the entire image or from regions. In the first case, the image is often characterized by its histogram thus obtaining a global image description. In the second case, image regions are obtained partitioning the image into tiles from which features are extracted; this is a way of representing the global features of the image at a finer resolution [1, 2]. The most important drawback to extract image visual content of both methods has been the inability to capture semantic content.

A better method to obtain regions is to use segmentation algorithms to divide images into homogeneous regions according to some criteria that discriminate between different entities of the image. This is the first step of all region-based image retrieval systems (RBIR). Then some descriptors are defined so that the retrieval can be performed [3, 4, 5, 6, 7]. These methods have significantly improved retrieval results, but they are still different from the results obtained by humans.

The main problem of current retrieval systems simulating the search performed by a human subject is the difference between human description of the queried image and the level of description (the extracted information) of retrieval system. Human subjects use high level concepts (and words) to identify elements of the image, actions or situations, whereas retrieval methods extract low level features (i.e. color, texture, shape, etc). The difference between these description levels is known as the 'semantic gap' [2]. One way to reduce the 'semantic gap', pointed out by Liu et al. [8] in their survey on CBIR systems, is the use of object ontology to define high-level concepts. This requires to obtain objects/entities of images. Some works have studied this issue in narrow application domains [9, 10, 11, 12]. Another way would be to define descriptors presenting the image components in linguistic terms, which is one of the goals of this paper.

Recently, the bag-of-words model uses image features as 'visual words' [13] of a wide vocabulary, mapped onto image categories by machine learning techniques [14]. The learning process deals with the whole width of the semantic gap. These approaches achieves important results in general categorization of scenes or objects even when the vocabulary is based on low-level features. One question that arise from our work is how these techniques could improve the results by introducing more semantic information in their vocabularies.

In the specific cases of color and texture, the most usual de-

*Fax: +34 977 559610
Email address: susana.alvarez@urv.es (Susana Alvarez)

scriptors are low-level features combined with shape or spatial location features. Descriptors are sometimes obtained from histograms [3, 15, 9, 16]. Other color descriptors capture the spatial color distributions: color layout (CLD) and color structure (CSD) descriptors. These last descriptors and descriptors obtained from histograms are included in the MPEG-7 [17] as standard color descriptors. In regard to texture descriptors, there are different sets of features, for example, wavelet features using Gabor filters [15, 18, 17, 7] or rotated complex wavelet filters [19], both define the multiscale descriptor as a vector containing energy and energy deviations before the corresponding filter is applied to the image. Liu and Picard [4] developed the 'Wold' features which distinguish between 'structured' and 'random' texture components. The former correspond to the peak magnitudes of image autocovariance and the latter are the MRSAR (Multiresolution simultaneous autoregressive model) estimated coefficients. Barcelos et al. [20] define a texture descriptor based on the modal matrix that represent the frequency space of an image consisting of eigenvectors that measure the proximity among points set of the quantized power spectrum of image. The modal matrix is their texture descriptor. Zhong and Jain [21]'s color and texture descriptor is a vector that contains some coefficients of the DCT (Discrete Cosine Transform) in JPEG image format. Lazebnik et al. [22] defined the *RIFT* descriptor as an sparse representation of the *SIFT* [23] that tries to cope with image textons assuring rotation invariance. All of these descriptors do not directly map the set of properties they extract to words describing the image.

If we focus on the problem of descriptors that can be mapped to real words, few descriptors have been developed. Most of them are generally related to color properties. Carson et al. [24] extracts two dominant colors from each region; Mojsilovic et al. [25] and Ma and Manjunath [5] from different codebooks, build feature vectors with the dominant colors and its corresponding occurrence percentage within the image. Smith and Chang [26], using a sparse binary vector representation of color sets, allow users to specify the color content within images by picking colors from a color chooser or by textual specification. Finally, Benavente et al. [27] proposed a fuzzy set model that directly maps colors to the eleven English basic color names. In the case of texture descriptors mapping words, Manjunath et al. [28] developed the PBC, which consist of three perceptual features: regularity, directionality and scale represented by bounded values. These features are related to the three most important perceptual dimensions in natural texture discrimination 'repetitiveness, directionality and granularity' identified by Rao and Lohse [29] in a psychophysical experiment. Recently, Salvatella and Vanrell [30] proposed a sparse texture descriptor that is based on describing texture through their blob attributes, this is the starting point of the proposal of this paper.

Focusing on the previous idea of mapping descriptors onto words we founded more recent works on image annotation [31, 32, 33, 34], these works follow a top-down methodology essentially based on machine learning techniques. The main focus relies on the accuracy on predicting good annotations by learning from previously annotated images, usually based on standard descriptors commonly used.

Here in this work we go back to the descriptor definition step by proposing a compact descriptor called *Texture Component Descriptor*, which deals with the annotation of color-textures without any learning step. Our descriptor relies on a pure bottom-up approach where feature selection is inspired on perceptual assumptions. We justify this backtracking to the descriptor definition because we can achieve two desired properties: the descriptor is low dimensional and comprehensive. This is, it is based on six dimensions with a direct perceptual correlation each. These properties can be achieved since we substitute machine learning effectiveness by strong perceptual assumptions. These are directly derived from the texton theory [35] which is complemented with perceptual grouping mechanisms capturing patterns emerging from the repetition of local attributes [36].

The paper is organized as follows: in section 2 we review the perceptual considerations justifying the attribute space where the descriptor is based on. In section 3 we propose a descriptor *Texture Component Descriptor* (TCD) derived from a 6D space that is an early fusion of a 3D blob space and a 3D color space. The next sections will explore the comprehensive nature of the proposed descriptor: in section 3.1 we propose a dense image representation for image segmentation, and in section 4 we define a grammar that translate our descriptor to basic linguistic terms that can improve it in browsing applications. Afterwards, section 5 compiles all the experiments that evaluates our approach. The first experiment demonstrates that our descriptor achieves similar performance to current best descriptors in retrieval; we compare our TCD to MPEG-7 in standard Corel datasets. Subsequent experiments explore the behavior of the descriptor from a qualitative point of view showing its feasibility in segmentation and browsing applications. In the last section we summarize the proposal and outline further work.

2. Texture and blobs

Texture representation has been the focus of a large amount of research in Psychophysics [37, 36, 35, 29] too. Two different schools of thought in the study of texture segregation have converged in their final conclusions. Both first-order statistics of local features and global spatial considerations are needed for a full representation. The present work is based on the texton theory of Julesz and Bergen [35] as the basis for the first steps in texture perception. After different conjectures, in 1983 Julesz proposed this theory that states three heuristics. First, texture discrimination is a preattentive visual task. Second, textons are the attributes of elongated blobs, terminators and crossings. Third, preattentive vision directs attentive vision to the location where differences in density of textons occur, ignoring positional relationships between textons. Finally, he gives an explicit example of textons in this way: *"elongated blobs of different widths or lengths are different textons"*. In summary, Texton theory concludes that preattentive texture discrimination is achieved by differences in first-order statistics of textons, which are defined as line-segments, blobs, crossings or terminators and their attributes: width, length, orientation and color.

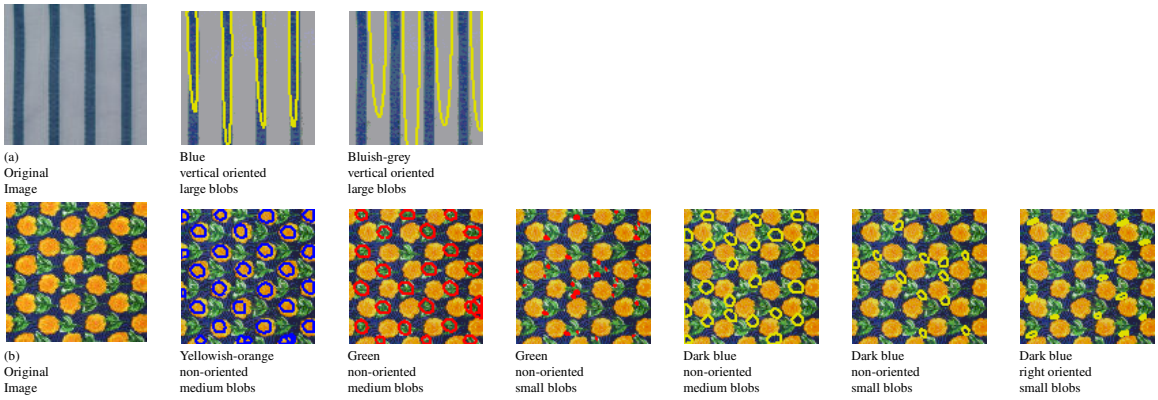


Figure 1: Textures components and their description

This perceptual theory is the consequence of an exhaustive study on local texture properties provoking preattentive texture discrimination in experimental conditions. In this work we propose to use these powerful results derived from a large psychophysical experimentation trying to prove different conjectures. These results allow us to substitute the usual training step on annotated image datasets of most computational approaches. Our hypothesis is based on the fact that these perceptual features can be encoding the efficiency of human visual representation. With the same goal, an early computational implementation of texton theory was done by Voorhees and Poggio [38], blob attributes on grey level images were used to determine boundaries between textures. In this work we propose to continue the work of Voorhees and Poggio [38] by updating it with recent computational operators [39] using color attributes [40] and inserting one further step that simulate a grouping mechanism onto the attributes that captures emergent repetitiveness [36].

Apart from the assumption that a texture can be described by their blob attributes, we also assume that a texture is provided by the existence of groups of similar blobs. This is the basis of the repetitiveness nature of texture images. Although a description based on blobs can be incomplete, the advantage of our proposal is that it gives a further step in reducing semantic gap. We are able to assign a basic semantic meaning to these blob low-level features by translating blob attributes to linguistic terms. Some examples of this proposal can be seen in Fig. 1. In image (a) a striped texture is described by two different types of blobs: blue elongated blobs and grey elongated blobs. In the same figure, texture (b) can be described in terms of 6 different types of blobs: blue, green and orange, of different sizes and shapes. The groups of blobs sharing similar features (size, orientation and color) is called *texture components*, and the texture description is obtained by joining the descriptions of these components. In the next sections we give a metric for this descriptor and a translation procedure to get the linguistic terms.

2.1. Blob components

To obtain the image's blob we use the same approach given in [30], which is based on the differential operators in the scale-space representation proposed by Lindeberg [41]. Assuming that image blobs have a Gaussian shape, we use the normalized differential Laplacian of Gaussian operator to detect the blobs of the image I,

$$\nabla^2_{norm} L_\sigma = \sigma^2 \nabla^2 L_\sigma \tag{1}$$

being $L_\sigma(I) = I * G(\cdot;\sigma)$. This operator also allows us to obtain the scale and the location of the blobs. The aspect-ratio and orientation of non-isotropic blobs are obtained from the eigenvectors and eigenvalues of the windowed second moment matrix [41].

To obtain the blob components of the color image, we apply the previously defined differential operators to the color channels. Since blob information emerge from both intensity and chromaticity variations, we use the opponent color representation that separates these two color dimensions. The first component of the opponent color space is intensity, $I = (R + G + B)/3$, and the other two are red-green, and blue-yellow chromaticity dimensions, which are given by

$$\begin{pmatrix} rg \\ by \end{pmatrix} = \begin{pmatrix} 1/2 & -1/2 & 0 \\ 1/3 & 1/3 & -2/3 \end{pmatrix} \cdot \begin{pmatrix} r \\ g \\ b \end{pmatrix} + \begin{pmatrix} 1/2 \\ 2/3 \end{pmatrix} \tag{2}$$

where r, g and b are normalized chromatic components, which are invariant to intensity changes and are given by $r = R/(R + G + B)$, $g = G/(R + G + B)$ and $b = B/(R + G + B)$. **R**, **G** and **B** are respectively the Red, Green and Blue components of the RGB space of the original image.

Detecting blobs on this opponent space implies some redundancy since a blob could be detected in any of the three channels. This redundancy has been eliminated by a perceptual filtering process, selecting the blobs of higher filter response ($\nabla^2_{norm} L_\sigma$) from those that overlap. Thus, this detection step provides us with a list of blobs and their attributes that we refer as Blob Components (BC), which are given in matrix form as:

$$\mathbf{B} = [\mathbf{B_{loc}} \mathbf{B_{sha}} \mathbf{B_{col}}] \tag{3}$$

where **B** is formed by joining three matrices: $\mathbf{B_{loc}}$ which is the location of the blobs, $\mathbf{B_{sha}}$ contains their shape attributes and

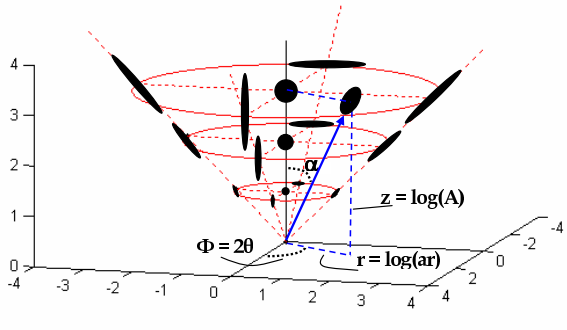


Figure 2: Shape-orientation blob space in cylindrical coordinates.

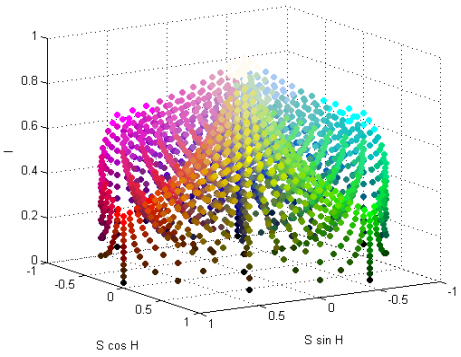


Figure 3: HSI color space.

$\mathbf{B_{col}}$ contains their color attributes. These matrices can be defined as:

$$\mathbf{B_{loc}} = [\mathbf{XY}] \tag{4}$$

where $\mathbf{X^T} = [x_1 \dots x_n]$, $\mathbf{Y^T} = [y_1 \dots y_n]$, being (x_j, y_j) the location of the center of j-th blob and n the number of blobs,

$$\mathbf{B_{sha}} = [\mathbf{WL\Theta}] \tag{5}$$

where $\mathbf{W^T} = [w_1 \dots w_n]$, $\mathbf{L^T} = [l_1 \dots l_n]$, $\mathbf{\Theta^T} = [\theta_1 \dots \theta_n]$ being (w_j, l_j, θ_j) shape attributes of the j-th blob (width, length and the orientation, respectively), and

$$\mathbf{B_{col}} = [\bar{\mathbf{I}}\,\overline{\mathbf{RG}}\,\overline{\mathbf{BY}}] \tag{6}$$

where $\bar{\mathbf{I}}^{\mathbf{T}} = [\bar{i}_1 \dots \bar{i}_n]$, $\overline{\mathbf{RG}}^{\mathbf{T}} = [\overline{rg}_1 \dots \overline{rg}_n]$, $\overline{\mathbf{BY}}^{\mathbf{T}} = [\overline{by}_1 \dots \overline{by}_n]$ being $(\bar{i}_j, \overline{rg}_j, \overline{by}_j)$ color attributes of the j-th blob (median of the intensity and chromaticity of the pixels that form the blob, respectively).

3. Textural Component descriptor

Once we have the Blob Components of the image we aim to group blobs with similar features (i.e. shape-orientation (w, l, θ) and color $(\bar{i}, \overline{rg}, \overline{by})$) in order to obtain the different *Textural Components* (TC) of the image. We use a uniform space similar to uniform spaces defined in color science to perform this grouping, where the distance between two points can be considered proportional to their perceptual difference. Since color and shape are independent features, we propose to use two different spaces to represent these blob attributes: one space to represent shape-orientation and another to represent the color of the blobs. The uniform three-dimensional space used to represent shape-orientation is similar to blob space as defined in [30]. They proposed a three dimensional cylindrical space where two axes represent the shape of the blob (aspect-ratio and area) and the third axis represents its orientation. The space we have used can be seen in Fig. 2.

The perceptual shape-orientation space is obtained by performing a non linear transformation U,

$$\begin{array}{rccc} U: & \mathbb{R}^3 & \rightarrow & \mathbb{R}^3 \\ & (w, l, \theta) & \rightarrow & (r, z, \phi) \end{array} \tag{7}$$

where $r = \log(ar), z = \log(A)$ and $\phi = 2\theta$, being ar the blob aspect ratio ($ar = w/l$), A its area ($area = w \cdot l$) and θ its orientation.
We should stress that in this space valid blobs are located inside the cone delimited by the angle $\alpha_{max} = \pi/4$ shown in figure 2, since this space defines the blob width as the shorter of the two lengths that characterize the blob.

The best color space to represent the color attributes of blobs would be the uniform and calibrated CIE-Lab space, but since the images we use are not calibrated we have chosen the HSI color space for two reasons: first, it is similar to uniform color spaces and has some correlation with the human perception of color and second, it is defined on cylindrical coordinates as the blob space defined above. This latter feature is interesting for the next process of grouping. We have used the color transform given in Gonzalez and Woods [42], where:

$$\begin{array}{rccc} HSI: & \mathbb{R}^3 & \rightarrow & \mathbb{R}^3 \\ & (R, G, B) & \rightarrow & (h, s, i) \end{array} \tag{8}$$

This space can be seen in Fig. 3. Color differences are computed as Euclidean distances in Cartesian coordinates $(s \cdot cos(h), s \cdot sin(h), i)$.

Considering the properties of these two spaces we state that similar blobs are placed on different unidimensional varieties, lines, rings or arcs. To group blobs of similar shapes and colors we use a clustering method that groups data with these points distributions and, at the same time, makes it possible to combine spaces with different characteristics, specifically color and shape-orientation. The clustering algorithm that has these properties is the Normalized Cut (N-cut) [43], which obtains the clusters by partitioning a graph in a recursive way, until the *N-cut* value exceeds a certain limit. This is the only parameter of the algorithm that determines the number of clusters obtained. In the graph, the nodes are the points of the feature space and the edges between the nodes have a weight equal to the similarity between nodes. A distance measure needs to be defined to determine the similarity between the nodes. Since the shape-orientation space has been designed to be uniform and the HSI color space is almost uniform, it is reasonable to use the Euclidean distance. Plataniotis and Venetsanopoulos [44] performs an experiment where they conclude that this distance used on the HSI color space is more discriminating, in color

difference, than Canberra and Minkowski's distance measures.

The N-Cut clustering algorithm can be defined as

$$NCUT([U(\mathbf{B_{sha}}), HSI(\mathbf{B_{col}})], \mathbf{\Omega}) = \{\hat{\mathbf{B}}^{\mathbf{1}}, \hat{\mathbf{B}}^{\mathbf{2}}, \ldots, \hat{\mathbf{B}}^{\mathbf{k}}\} \quad (9)$$

where, $\mathbf{\Omega}$ is the weight matrix, and its elements define the similarity between two nodes through the calculation of the distance in each one of the spaces (shape-orientation and color) in an independent way. These weights are defined as,

$$\begin{aligned} \omega_{pq} &= \exp - \frac{\|U(\mathbf{B_{sha}})_p - U(\mathbf{B_{sha}})_q\|_2^2}{\sigma_{sha}^2} \\ &\cdot \exp - \frac{\|HSI(\mathbf{B_{col}})_p - HSI(\mathbf{B_{col}})_q\|_2^2}{\sigma_{col}^2} \end{aligned} \quad (10)$$

This weight represents the similarity between blob p and blob q that depends on the similarity of its shape-orientation features and the similarity of its color features. $U(\mathbf{B_{sha}})_p$ and $HSI(\mathbf{B_{col}})_p$ are the p-th row of the matrices $U(\mathbf{B_{sha}})$ and $HSI(\mathbf{B_{col}})$ respectively. Shi and Malik [43] defined σ as the percentage of the total range of the feature distance function. In our case, σ_{sha} is defined in the shape-orientation space and σ_{col} in the color space. We have empirically fixed these values as the 12 and 16 % of the range of the shape-orientation and color distance respectively.

The result of the clustering obtained by the N-cut algorithm is represented by $\hat{\mathbf{B}}^{\mathbf{i}}$, $\forall i = 1, ..., k$ (where k is the total number of clusters). It is the i-th class of the clustering process that will be the i-th texture component, this comprises,

$$\hat{\mathbf{B}}^{\mathbf{i}} = [\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{loc}} \hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{sha}} \hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{col}}] \quad (11)$$

where $\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{loc}} = [X^i Y^i]$, $\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{sha}} = [R^i Z^i \Phi^i]$ and $\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{col}} = [H^i S^i I^i]$, being $X^i \subset \mathbf{X}$ a subset of $\mathbf{X}$ defined by those elements belonging to cluster i and the rest of the terms Y^i, R^i, Z^i, Φ^i, H^i, S^i and I^i are defined in an equivalent way.

These clusters of blobs with similar attributes, $\hat{\mathbf{B}}^{\mathbf{i}}$, are the basis for our descriptor, named *Texture Component Descriptor* (TCD), that for a given image it is denoted as

$$TCD = \{TCD^1, \ldots, TCD^i, \ldots, TCD^k\} \quad (12)$$

where each TCD^i is given by the blob attributes of the prototype for each texture component or cluster. This prototype is computed by estimating the median of all the blob attributes in that cluster, $[\hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{sha}} \hat{\mathbf{B}}^{\mathbf{i}}_{\mathbf{col}}]$. This give a 6-dimensional description for each cluster or TC:

$$TCD^i = (r^i, z^i, \phi^i, h^i, s^i, i^i) \quad (13)$$

In this way the descriptors of an image are the shape-orientation (3D) and color attributes (3D) of its TC.

3.1. Dense representation

Once the image has been decomposed in its textural components (set of blobs with similar characteristics, see Fig. 4), we build the dense image representation. To this end, we expand the properties of each textural component ($\hat{\mathbf{B}}^{\mathbf{i}}$) to all the pixels in the influence area of its blobs. This influence area is the image region containing the group of blobs that form the texture component.

To obtain the influence area of a textural component, we estimate the periodic distance between its blobs since we know the spatial location of blob centres. The maximum frequency of the blob distance matrix provide this period estimation for each textural component,

$$p_i = \arg\max_d(Hist(\mathbf{DT}^{\mathbf{i}})), \forall i = 1, 2, \ldots k \quad (14)$$

where $d \in \mathbf{DT}^{\mathbf{i}}$, that is the distance matrix between all blob centres of the i-th texture component, $\hat{B}^i_{loc}$, and $Hist$ is the histogram.

To assign an image pixel to a textural component from its detected blobs we build a binary image $\mathbf{I}_{\hat{\mathbf{B}}^{\mathbf{i}}}$ where pixels belonging to the detected blobs are set to one. Afterwards, we perform a morphological closing operation [45] to expand the blob properties to all the points in its influence area which is given by the estimated period, this is

$$\mathbf{I}_{\mathbf{C}^{\mathbf{i}}} = ((\mathbf{I}_{\hat{\mathbf{B}}^{\mathbf{i}}} \oplus EE_{p_i}) \ominus EE_{p_i}), \forall i = 1, 2, ...k \quad (15)$$

where EE_{p_i} is a circular structuring element with radius $p_i/2$, that creates compact regions containing blobs with similar shapes, orientations and colors, and their neighboring pixels. The radius of the structuring element has coped with the spatial structure derived from the period of its blobs. The expansion of the blob properties is inspired in the intracortical inhibition step of Malik and Perona [46].

In this way, we obtain a k-dimensional image representation (being k the number of textural components of the image) that is our blob-based dense image representation (BR). Every pixel of this image representation is given by a feature vector of k components (being every component a binary value), which represents the membership to a specific texture component, given by its TCD descriptor. In the bottom part of the Fig.4 we show an example of a pixel representation.

4. A basic color-texture vocabulary

In this work we take a first step towards the construction of a vocabulary of basic terms in natural language, for color texture. We propose to use plain English words to describe geometry and photometry of the image blobs. Since a texture is described by a list of texture components each defined by their blob attributes, we can build one linguistic phrase to describe these attributes. Thus a complete description of a color texture is given by a list of phrases explaining the texture parts. Although this description does not bridge all the semantic gap, it gives an important step in providing semantic properties that is new in the frame of texture research.

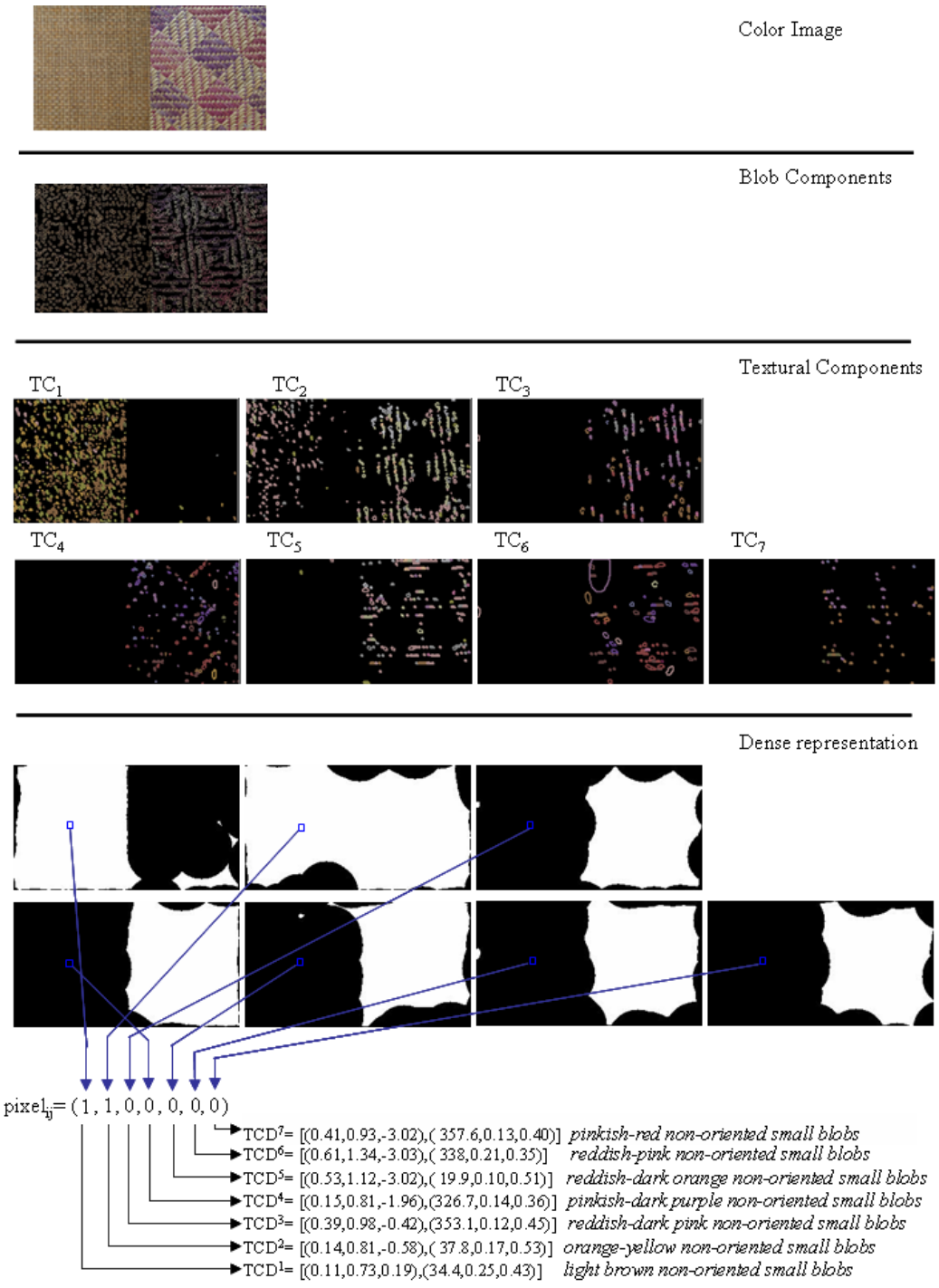


Figure 4: The stages of the blob-based dense image representation proposed.

Following this we detail a procedure for an automatic translation of the TCD descriptor (given in equation 12 for texture components) to phrases. To do this, we first introduce the basic terms we use as vocabularies for each blob attribute,

Color 11 terms defined by Berlin and Kay [47] and modeled by Benavente et al. [27]. Moreover, we will use the same 11 basic terms with the *-ish* modifier.

Intensity 2 terms (Dark or Light) to modify the basic terms of color. They are computed on the intensity component and are specific for each basic color.

Shape 2 terms to describe shape of blobs, which are non-oriented, to refer to isotropic or near isotropic blobs, and oriented to refer to elongated blobs.

Size 3 terms to describe size of blobs: small, medium and large.

Orientation 4 different modifiers to describe the orientation of the elongated blobs; we have simplified them by using the following terms: horizontal ($\approx 0°$), right ($\approx 45°$), vertical ($\approx 90°$) and left ($\approx 135°$).

Second, we give a syntax to systematically translate from texture component to phrases. It is given by the following grammar, in BNF (*Backus Naur Form*), and using previous vocabularies:

Texture_description := Texture_component [',' Texture_description]
Texture_component := Color_description Shape_description 'blobs'
Color_description := [*'Dark '* | *'Light '*]
  Basic_term |Basic_termish–Basic_term |Basic_term–Basic_term
Basic_term := *'red'* | *'orange'* | *'brown'* | *'yellow'* | *'green'* | *'blue'*
  | *'purple'* | *'pink'* | *'black'* | *'grey'* | *'white'*
Basic_termish := *'reddish'* | *'orangish'* | *'brownish'* | *'yellowish'*
  | *'greenish'* | *'bluish'* | *'purplish'* | *'pinkish'* | *'blackish'*
  | *'greyish'* | *'whitish'*
Shape_description := Orientation_description Size_description
Orientation_description := [*'Non–Oriented'* |
  Basic_orientation *'Oriented'*]
Basic_orientation := *'horizontal'* | *'right'* | *'vertical'* | *'left'*
Size_description := *'small'* | *'medium'* | *'large'*

To select the terms from the values of the TCD we have used different criteria. For color description we assign names based on the fuzzy system defined by Benavente et al. [48]. In this frame the 11 basic terms are parameterized by sigmoid functions that assign membership values to each color term. A unique color term is assigned if its membership is high, that is, we consider it a pure color. For non pure colors we use just the first two greatest memberships, e.g. colors are in boundaries of just two color terms and therefore a bi-lexemic term is used (hyphen form). If one of the two memberships is still predominant we use the *-ish* modifier for the non-predominant color, otherwise we use the two basic terms. Moreover, color description can be modified by an intensity term as *dark* or *light*, in this case the term refers to the position of the color in the intensity axis. Dark modifiers are assigned to intensities over the 90% of the color intensity, and Light modifiers are for intensities under the 10%.

For shape description we have used highly simplified vocabularies. Shape is constrained to two simple forms of blobs, orientation is sampled to four terms, and size is reduced to three. With regard to size, our descriptor is not scale-invariant (blob areas are computed in pixels). Therefore, the assignment of size terms will be dependent on image size and this is an important point to be considered to form queries. Size specified in a query has to be adjusted to the relative size of the pattern within the image.

Thus this previous grammar with the introduced criteria provide impossible color combinations (such as *whitish-white* or *dark black*). After removing these useless color descriptions we have a semantic dictionary of approximately 2085 phrases to explain texture components. Several examples of these descriptions are shown in figures 1, 4, 10 and 11.

5. Experiments

To evaluate our descriptor we have done three different experiments. In Experiment 1 we test its efficiency in coping similarity in an image retrieval application, in Experiment 2, we prove the feasibility of its dense representation to be used to locate textures in images, and in Experiment 3, we do a qualitative exploration of the proposed vocabulary that can be derived from the descriptor.

To perform the experiments we have built our specific dataset of images coming from different databases. The selection of images was based on the criteria of having homogeneous textures to preserve the same appearance in any subimage of the given image. With this property we assure that errors in retrieval would only be due to the descriptor and not to inhomogeneous properties of the texture images (such us inhomogeneous lighting or background).

Next, we detail the sources and how the dataset has been built:

- Mayang's Texture database[1]. 59 images have been selected, they can be seen in Fig. 5(a). We extracted 354 subimages by taking 6 sub-images of every image.

- Outex database [49]. 6 images randomly selected from each one of the 16 original textures (in Fig. 5(b), extracted from Outex-TC-00013) obtaining 96 images for our dataset.

- VisTex database[2]. We selected 5 images (shown in Fig. 5(c)), commonly used in these experiments and we obtained 30 images for our dataset.

[1]Dataset by W. Smith and A.M. Murni, 2001. http://www.mayang.com/textures/

[2]MIT Media Lab, Vision Texture-VisTex database, 1995. http://vismod.media.mit.edu/vismod/imagery/VisionTexture/vistex.html

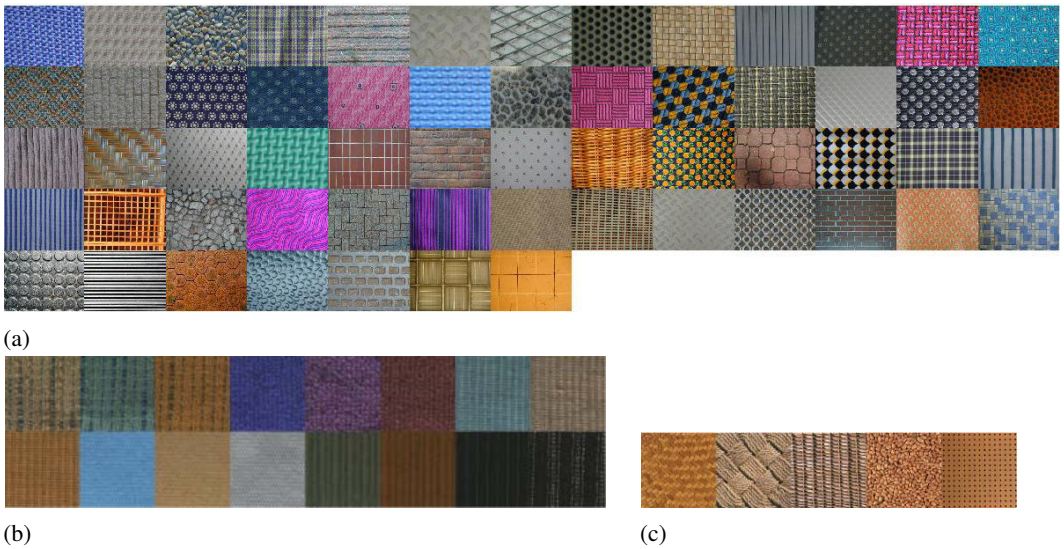


Figure 5: Images in our dataset. (a) From Mayang database. (b) From the Outex database. (c) From VisTex database.

Thus, the dataset has 480 images (the resolution of the samples is 100 x 100 pixels) of 80 textures. We stress that whereas all images in the Outex database were acquired in an strictly controlled environment (known illumination sources and imaging geometry), the rest of the images were acquired under non controlled conditions.

5.1. Experiment 1

In this experiment we evaluate the performance of *TCD* descriptor in image retrieval. In order to do so we compute the distance between images and we need to use an adequate similarity measure bearing the following consideration: images can have different number of Textural Components (TC) (see equation 12) where the number of components (or clusters) depends on the complexity of image content, this is automatically determined by the clustering algorithm. We have used the Earth Mover's Distance (EMD) [50] that fulfills this consideration. This metric requires to define a ground distance between two clusters. In our case this corresponds to the distance between two components of a *TCD*. Below we define this ground distance by combining with a weighting parameters the two feature spaces (shape-orientation and color) in the following manner:

$$d(TCD^i, TCD^j) = \alpha \cdot d_{shape}(TCD^i, TCD^j) + \beta \cdot d_{color}(TCD^i, TCD^j)$$

where d_{shape} and d_{color} are Euclidean distances in the shape-orientation space and color space, respectively, that are derived from the uniform property of both spaces. Each distance has been normalized; this is possible because our feature spaces are bounded independently of the image content. Shape-orientation space has the limits of blob attributes and color space is bounded by the maximum luminance. The parameters α and β are weighting these two distances. To obtain a good estimation of these two parameters it would be necessary to perform additional psychophysical experiments. Some studies have tackled the problem of combining texture and color [51] concluding that weights applied to color and texture when estimating similarities are highly dependent on the observer and the context.

Retrieval experiments have been performed on our dataset introduced above (figure 5) and three sets of Texture images from the Corel stock photography collection[3]. Textures (137000), Textures II (404000) and Various Textures I (593000). In the experiment we refer to them as *Corel*, *Corel2* and *Corel1* respectively. Each Corel group has 100 textures (768 x 512 pixels), every texture is divided into 6 subimages and the total number of textures is $6x100 = 600$ for each Corel dataset. In figure 6 we show some textures of the three Corel datasets.

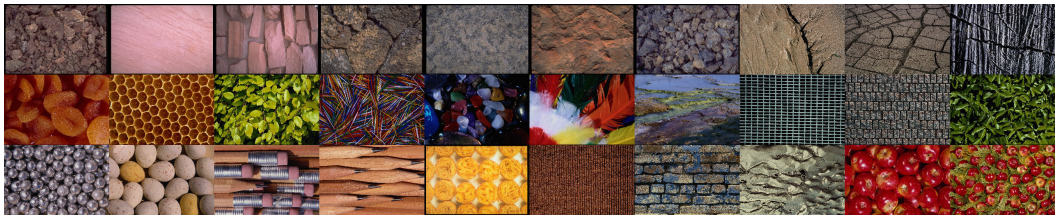
We have used the almost standard Recall measure [52] to evaluate the performance of the retrieval and the precision-recall curves. These measures have been computed by using all the images of each dataset as query images and afterwards we have computed the average of all queries. In the ideal case of the retrieval, the top 6 retrieved images would be from the same original subsampled image.

Using similar weights in the combination of shape and color descriptors to compute the distance (α, β in equation 5.1) we have found that they do not have a big influence in the average recall measure. This is probably due to the fact that color and texture information is integrated earlier, at the blob level, before building the descriptor *TCD*. This fact is illustrated in Fig.7 for our dataset (a) and for *Corel* dataset in (b). Best results in all datasets are obtained when both color and shape are combined, otherwise average rate decreases substantially.

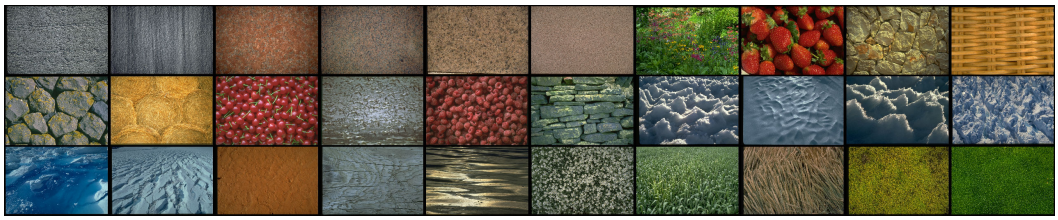
To compare efficiency, in table 1 we show retrieval rates for the 4 datasets using our *TCD* and two MPEG-7 descriptors [53]. We have combined two MPEG-7 descriptors, HTD (*Homogeneous Texture Descriptor*) and SCD (*Scalable Color Descrip-*

[3]Corel data are distributed through http://www.emsps.com/photocd/corelcds.htm

(a) *Corel*

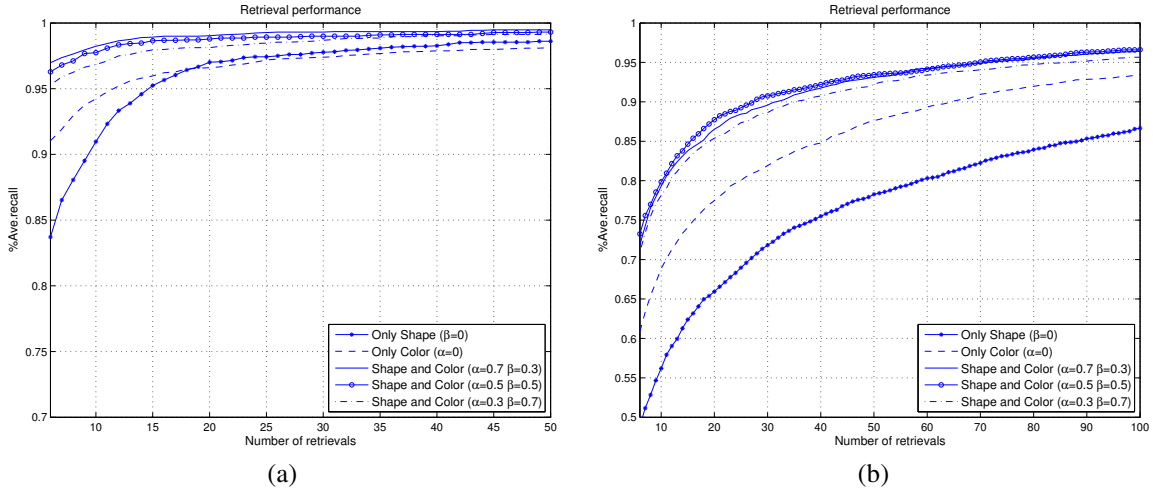


(b) *Corel1*



(c) *Corel2*

Figure 6: Corel datasets.



(a) (b)

Figure 7: Retrieval performance of TCD with different weights. (a) Our dataset. (b) *Corel* dataset.

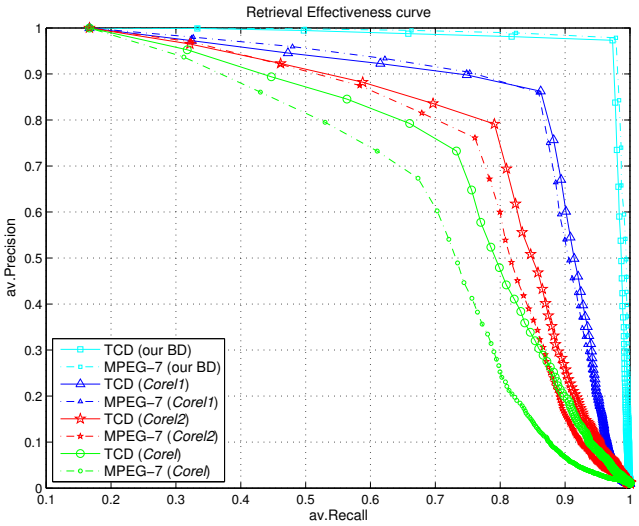


Figure 8: Precision-Recall curves of TCD and MPEG7 descriptors (HTD and SCD) for different datasets.

.

Descriptor	our dataset	*Corel*	*Corel1*	*Corel2*
TCD	97.33	**73.25**	**86.25**	**79.11**
MPEG-7 (SCD & HTD)	**97.85**	67.33	85.94	76.11

Table 1: Average Retrieval Rate

tor) as they are combined in Dorairaj and Namuduri [15]. The Average Retrieval rate computed shows how our *TCD* overcome MPEG-7 descriptors for the three Corel datasets and a similar performance is achieved for our dataset. We also show precision-recall curves using the same datasets in Fig.8, where we show the full evaluation that confirms the previous results over the precision range.

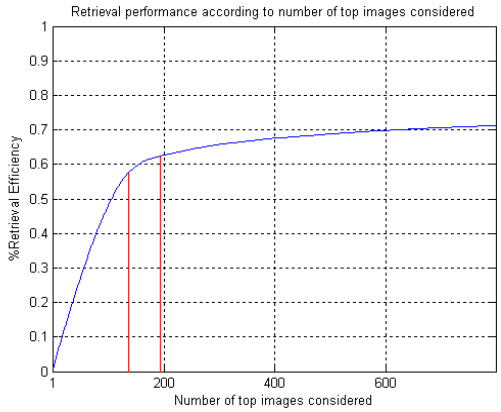


Figure 9: Retrieval performance using a BR-based segmentation.

5.2. Experiment 2

In this experiment we evaluate the dense representation in a retrieval application that is based on a weak segmentation. This allows us to evaluate its efficiency with images containing different textural patterns.

Assuming that a texture is formed by several textural components spatially grouped on the same region, we perform the image segmentation by clustering the feature vectors of the representation build in section 3.1. We have used a SOM (Self Organizing Map) neural network [54] to perform the clustering, although we could have used any other simple clustering technique. This process generates N regions associated to each one of the textural patterns of the image, where N is defined by the user. Considering an image as a mosaic of different texture regions [18], we have built a dataset formed by 1500 mosaic images, were each mosaic is composed by selecting 9 random images from our dataset with the restriction that no mosaic have a repeated texture.

Before performing the image retrieval, all the mosaics of the image dataset are segmented and decomposed into several regions using the proposed dense representation. Then in each region we compute its descriptor (TCD) inside the biggest inscribed rectangle of the region. For a given image query we compute its distance to each one of the image regions, of all dataset's images.

The image queries used in the retrieval are the same as those used in the first experiment. The effectiveness of the retrieval is again evaluated computing the Retrieval rate. In the ideal case all the top N retrievals belong to the same original large image. However, since the texture images that compose the mosaics has been randomly chosen, there is not a unique value of N. Concretely we have between 137 and 194 images of the same texture through the several mosaics. This implies that in Fig. 9 is not possible to mark the point where we expect to achieve the maximum possible efficiency (the top N retrievals), so we marked with red lines these 137 and 194 values. In this figure we can see that the retrieval efficiency is between 57% and 62%. In this way, we can quantify the reduction of efficiency that is due to the segmentation process.

Although the reduction is important, an efficiency ≈60% is still an acceptable rate for a representation that locates textures in images. This is an important issue if we want to evaluate the applicability of the proposed descriptor. A descriptor with semantic properties but without the ability of locating textures in images could not be completely justified.

In Figures 10 and 11 we show two examples of image queries and their corresponding retrieval results. For every image we give its ranking in the retrieval. In both cases we show the first 5 retrieval results and the subregion (white rectangle) where it was detected, in the 2nd row we show the first five failed results. A qualitative analysis of the results make us to highlight two points, first the boundaries of segmentation need to be improved mainly when the size of blobs increases. Second, in most of the examples the errors arise from images that are quite similar to the query, following the properties of the metric described in previous experiment.

To conclude this section we have added a similar retrieval experiment but with natural images instead of squared mosaics. We have built a small dataset of 21 images (see Fig.12). We show the results from three different queries extracted from the same images. Top five retrieved images and the regions where they were found are shown in figure 13. Although some further research needs to be done to improve segmentation, we again conclude the applicability of the proposed descriptor and its dense versions for general natural images.

5.3. Experiment 3

From a qualitative point of view we analyzed how the appearance of textures is represented by the proposed descriptor using Multidimensional Scaling (MDS) [55] in order to reduce the dimension of the representation. In figure 14 we show a global plot of the MDS computed on the distance matrix obtained in Experiment 1, the stress measure obtained for our dataset is 0.165 (the ideal value is 0 which indicates a perfect low dimension representation). In this figure we can see the combination of the two cylindrical spaces, color and shape-orientation. In the center we see an important overlapping of images, whereas in the external area we can see several examples of dominant properties. It seems that in the area around the circumference we have set there are images with dominant saturated colors and dominant directions of anisotropic blobs. Furthermore, along circular axis of dominant properties we can see how the hues (blue, pink-purple, brown-red) are grouped. In figure 15 we show a zoom of the central part of this plot where we can see three different types of images: textures with a dominant color but with low saturation (greyish patterns), textures with isotropic blobs, and textures with properties of the extreme axis but appearing together on the same image, that is two or more saturated colors of different hues (e.g. brown and blue), or extreme orientations on the same pattern (e.g. $90 - 0$).

As an example of browsing, instead of using sub-images as a query, in figure 16 we show two examples of queries formulated in terms of our vocabulary. In both figures we show the 2D plot of MDS computed on the distance matrix of the descriptors of top 40 images retrieved from a textual query. At the top of this figure we show the result from the query: *"Blue vertical oriented large blobs"*, showing with a frame Q the position of the query descriptor. In this case two essential axis emerge, one for the orientation and another for the color saturation, as it is provided by the query. The figure at the bottom is the result of the more complex query *Blue horizontal oriented small blobs AND Brown horizontal oriented small blobs*, which is based on two different texture components. The configuration obtained on a 2D plot of MDS clearly shows an axis where the two colors of the query appear on the extremes. In the center and next to the query position we can see the images where both colors appear together. In this 2D plot there is not a clear interpretation for shape.

At this point we have to conclude that this is a preliminary and qualitative analysis to illustrate the behavior of the metric for the proposed descriptor, which seems to be coherent with the texture appearance explained with basic linguistic terms.

6. Conclusions

This paper proposes a computational approach implementing a perceptual theory that combines color and texture in a early fusion way using a low-dimensional space that copes with blob attributes. It provides a comprehensive framework since it allows to define sparse, dense and linguistic descriptions of color texture images.

The work implements the original definition of the Julesz's perceptual theory Julesz and Bergen [35], where textons are essentially defined by the attributes of image blobs. The attributes we propose are: area, aspect-ratio and orientation (for shape), and color, that defines a low dimensional color-texture space.

We propose a color-texture descriptor: the Texture Component Descriptor (TCD), that arise from the decomposition of the image in its *textural components*, which are groups of blobs with similar attributes either color, shape or orientation. This is based on a clustering on the perceptual spaces of the blob attributes. Clusters of blobs are coping with the inherent repetitive property of the image texture. We compare our proposed descriptor with a combination of two MPEG-7 descriptors (HTD and SCD) in a retrieval experiment. Our descriptor overcomes MPEG-7 in three Corel datasets of natural textures.

Moreover, we present two additional experiments that explore the comprehensive qualities of the proposed framework. First, we show that the descriptor can be extended to a dense representation inspired on a winner-take-all mechanism computed with morphological operations. This color-texture representation shows a reasonable performance in locating texture patterns included in complex images. Second, we give a procedure to translate the descriptor to a preliminary vocabulary based on basic English terms. The experiment gives a qualitative evaluation of the proposed vocabulary using Multidimensional Scaling to explore the perceptual properties of the descriptor on the whole image dataset. Additionally, we plot some examples of the retrieved images from term-based queries that show the feasibility of the descriptor in browsing applications.

Further work is needed to evaluate the performance of our proposal in larger datasets. Introducing structural properties of

Green non-oriented medium blobs
Dark blue non-oriented medium blobs
Yellowish-orange non-oriented medium blobs

Dark blue non-oriented small blobs
Dark blue right oriented small blobs
Green non-oriented small blobs

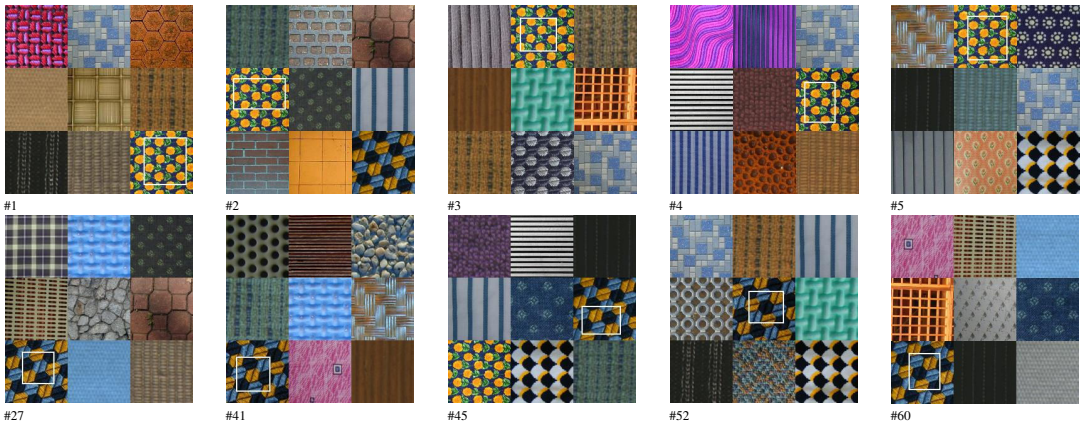


Figure 10: Query image and its textual description (on the top). 1st row: top 5 retrieved images. 2nd row: top 5 errors in retrieval.

Grey non-oriented small blobs
Grey horizontal oriented small blobs
Grey vertical oriented small blobs

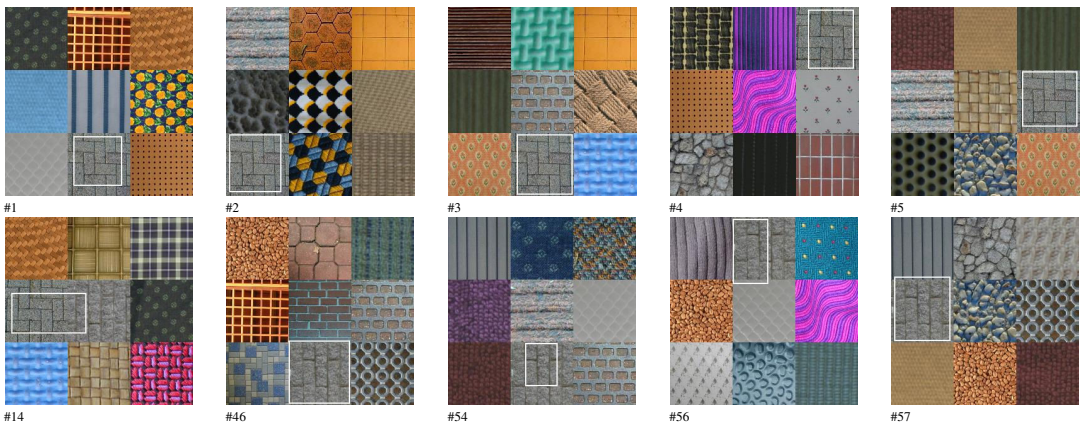


Figure 11: Query image and its textual description (on the top). 1st row: top 5 retrieved images. 2nd row: top 5 errors in retrieval.

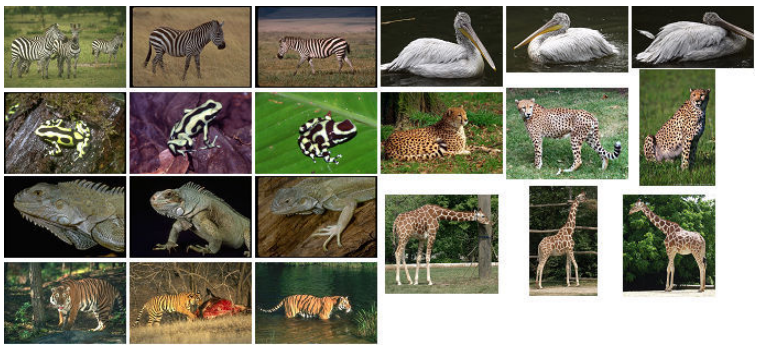


Figure 12: Natural images Dataset.

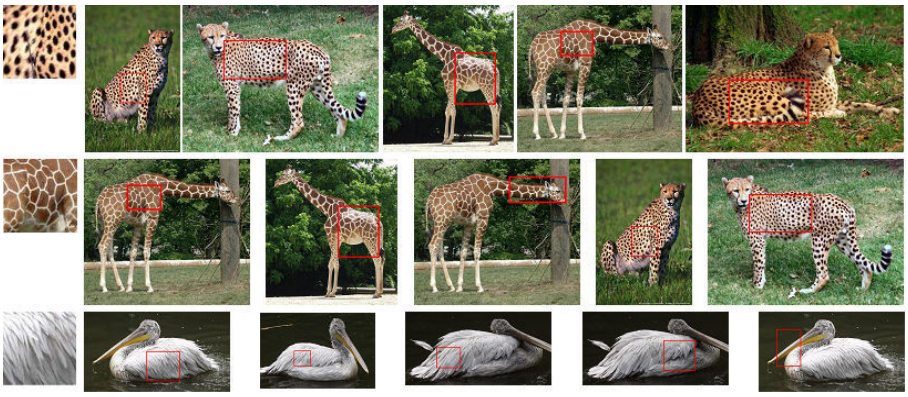


Figure 13: Top 5 retrieved images of the query on the left column.

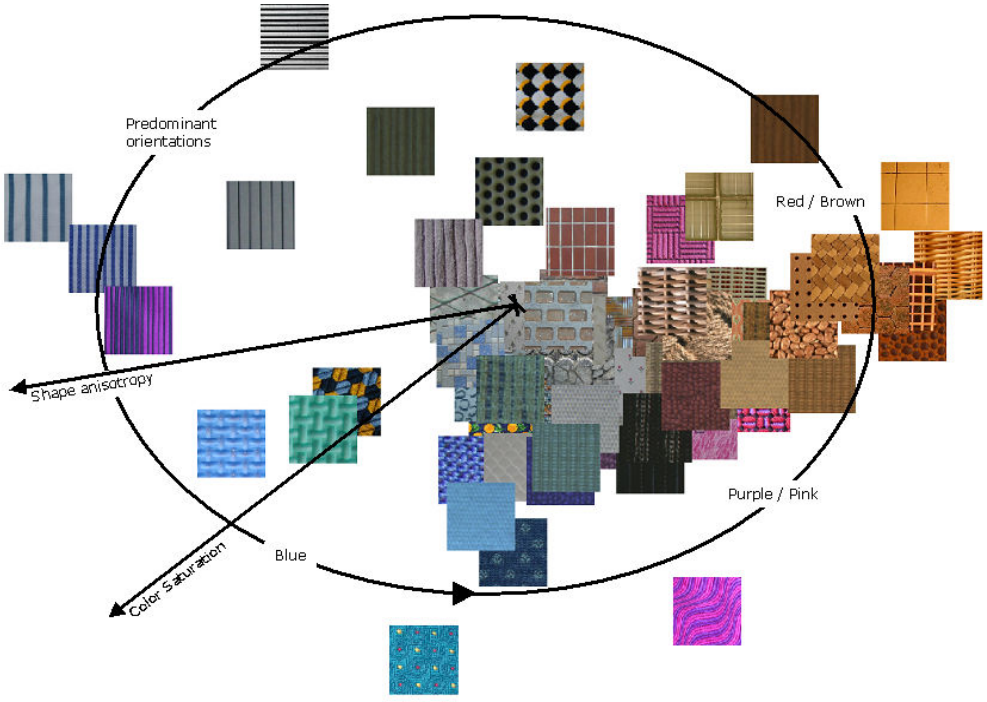


Figure 14: 2D MDS configuration of the our image dataset.

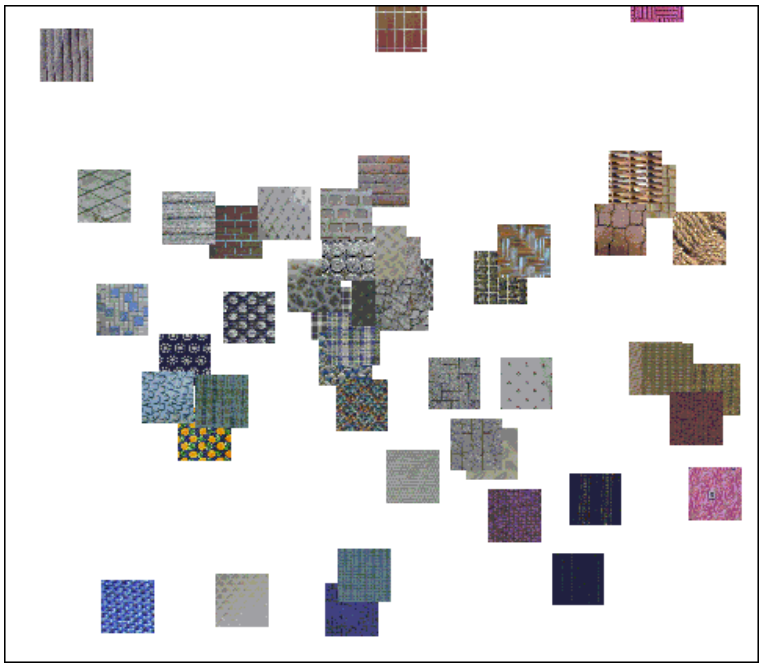


Figure 15: Central zoom of the 2D MDS results.

the texture patterns that emerge from the blob organization (e.g. regularity), using the localization of the blobs that is already computed with texture components, could clearly improve the descriptor for browsing.

7. Acknowledgments

This work has been partially supported by projects TIN2007-64577, TIN2010-21771-C02-1 and Consolider-Ingenio 2010 CDS2007-35100018 of Spanish MICINN (Ministry of Science and Innovation).

References

[1] F. Long, H. Zhang, D. Feng, Multimedia Information Retrieval and Management, Springer.

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

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

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

[5] W. Ma, B. Manjunath, NeTra: A toolbox for navigating large image databases, in: Proceedings of the International Conference on Image Processing (ICIP), volume 1, pp. 568–571.

[6] B. Prasad, K. Biswas, S. Gupta, Region-based image retrieval using integrated color, shape, and location index, Computer Vision & Image Understanding 94 (2004) 193–233.

[7] J. Wang, J. Li, G. Wiederhold, SIMPLIcity: Semantics-sensitive integrated matching for picture libraries, IEEE Transaction on Pattern Analysis and Machine Intelligence 23 (2001) 947–963.

[8] Y. Liu, D. Zhang, G. Lu, W. Ma, A survey of content-based image retrieval with high-level semantics, Pattern Recognition 40 (2007) 262–282. Good.

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

[10] C. Fuh, S. Cho, K. Essig, Hierarchical color image region segmentation for content-based image retrieval system, IEEE Transactions on Image Processing 9 (2000) 156–162.

[11] T. Gevers, A. Smeulders, PicToSeek: combining color and shape invariant features for image retrieval, IEEE Transactions on Image Processing 9 (2000) 102–119.

[12] A. Pentland, R. Picard, S. Sclaroff, Photobook: tools for content-base manipulation of image databases, Technical Report 255, MIT Media Laboratory Perceptual Computing, 1993.

[13] J. Sivic, B. Russell, A. Efros, A. Zisserman, W. Freeman, Discovering objects and their localization in images, in: Proc. IEEE Int. Conf. on Computer Vision (ICCV), volume 1, IEEE Computer Society, Los Alamitos, CA, USA, 2005, pp. 370–377.

[14] L. Fei-Fei, P. Perona, A bayesian hierarchical model for learning natural scene categories, in: CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), volume 2, IEEE Computer Society, Washington, DC, USA, 2005, pp. 524–531.

[15] R. Dorairaj, K. Namuduri, Compact combination of MPEG-7 color and texture descriptors for image retrieval, in: Conference on Signals, Systems and Computers. Conference Record of the Thirty-Eighth Asilomar, volume 1, pp. 387–391.

[16] J. Smith, S. Chang, Local color and texture extraction and spatial query, in: Proc. of IEEE Int. Conf. on Image Processing, volume 3, pp. 1011–1014.

[17] B. Manjunath, J. Ohm, V. Vinod, A. Yamada, Color and texture descriptors, IEEE Trans. Circuits and Systems for Video Technology, Special Issue on MPEG-7 11 (2001) 703–715.

[18] B. Manjunath, W. Ma, Texture features for browsing and retrieval of image data, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (1996) 837–842. Good, basic for texture features.

[19] M. Kokare, P. Biswas, B. Chatterji, Texture image retrieval using new rotated complex wavelet filters, IEEE Transactions on Systems, Man, and Cybernetics- Part B 35 (2005) 1168–1178.

[20] C. Z. Barcelos, M. Ferreira, M. Rodrigues, Retrieval of textured images through the use of quantization and modal analysis, Pattern Recognition 40 (2007) 1195–1206.

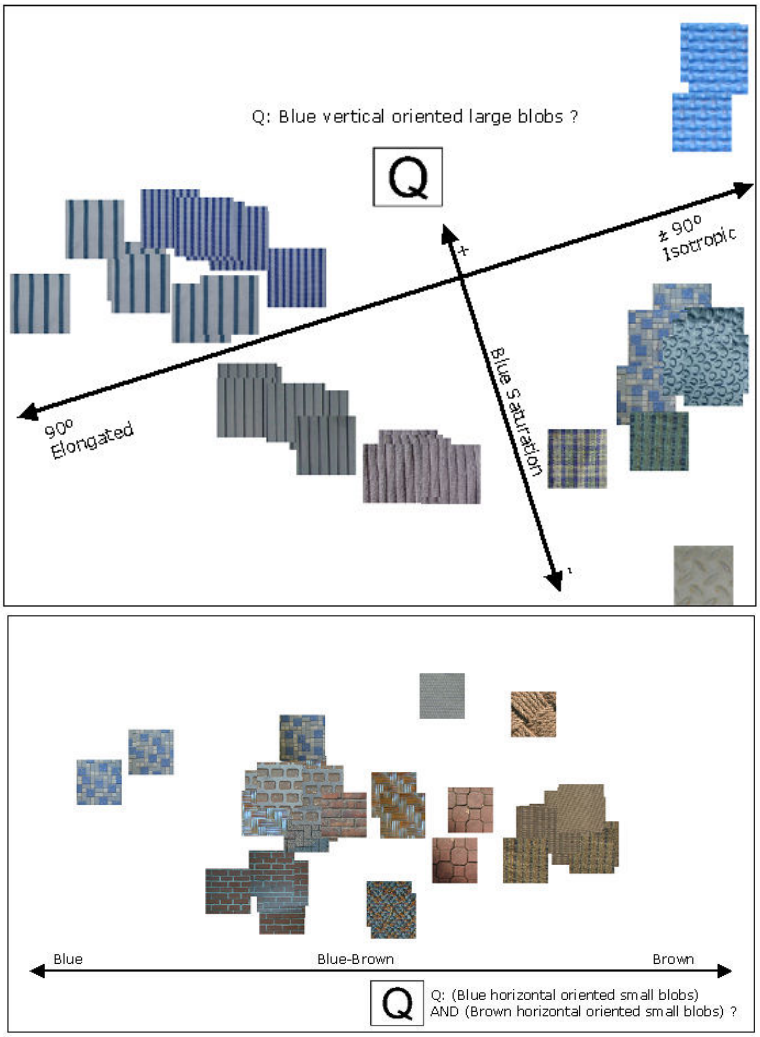


Figure 16: 2D MDS configuration of top 40 images retrieved using a textual query denoted as Q in the plot.

[21] Y. Zhong, A. K. Jain, Object localization using color, texture and shape, Pattern Recognition 33 (2000) 671–684.
[22] S. Lazebnik, C. Schmid, J. Ponce, A sparse texture representation using local affine regions, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2005) 1265–1278.
[23] D. Lowe, Distinctive images features from scale-invariant keypoints, Int. Journal of Computer Vision 60 (2004) 91–110.
[24] C. Carson, S. Belongie, H. Greenspan, J. Malik, Region-based image querying, in: CVPR'97 Workshop on Content-Based Access of Image and Video Libraries, pp. 42–49.
[25] A. Mojsilovic, J. Kovacevic, J. Hu, R. Safranek, S. Ganapathy, Matching and retrieval based on the vocabulary and grammar of color patterns, IEEE Transactions on Image Processing 9 (2000) 38–53. Good.
[26] J. Smith, S. Chang, Single color extraction and image query, in: Proc. of IEEE Int. Conf. of Image Processing, volume 3, Washington, pp. 528–531.
[27] R. Benavente, M. Vanrell, R. Baldrich, A data set for fuzzy colour naming, Color Research and Application 31 (2006) 48–56.
[28] B. Manjunath, P. Wu, S. Newsam, H. D. Shin, A texture descriptor for browsing and similarity retrieval, Journal of Signal Processing: Image Communication 16 (2000) 33–43.
[29] A. Rao, G. Lohse, Towards a texture naming system: Identifying relevant dimensions of texture, Vision Research 36 (1996) 1649–1669.
[30] A. Salvatella, M. Vanrell, Blob detection and grouping for texture description and other applications, Technical Report 110, Computer Vision Center, Bellatera Autonoma University, 2007.
[31] V. Lavrenko, R. Manmatha, J. Jeon, A model for learning the semantics of pictures, in: Advances in Neural Information Processing Systems 16 (NIPS), pp. 553–560.
[32] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. M. Blei, M. I. Jordan, Matching words and pictures, Journal Maching Learning Research 3 (2003) 1107–1135.
[33] R. Datta, W. Ge, J. Li, J. Z. Wang, Toward bridging the annotation-retrieval gap in image search, IEEE MultiMedia 14 (2007) 24–35.
[34] N. Rasiwasia, N. Vasconcelos, Holistic context modeling using semantic co-occurrences, Conf. Computer Vision and Pattern Recognition (CVPR) (2009) 1889–1895.
[35] B. Julesz, J. Bergen, Textons, the fundamental elements in preattentive vision and perception of textures, Bell Systems Technological Journal 62 (1983) 1619–1645.
[36] J. Beck, A. Sutter, R. Ivry, Spatial frequency channels and perceptual grouping in texture segregation, Comput. Vision Graph. Image Processing 37 (1987) 299–325.
[37] J. Bergen, Theories of visual texture perception, in: D. Regan (Ed.), Vision and Visual Dysfunction, volume 10B, New York - MacMillan, 1991, pp. 114–134.
[38] H. Voorhees, T. Poggio, Computing texture boundaries from images, Nature 333 (1988) 364–367.
[39] T. Lindeberg, Feature detection with automatic scale selection, Int. Journal of Computer Vision 30 (1998) 79–116.
[40] S. Alvarez, A. Salvatella, M. Vanrell, X. Otazu, 3d texton spaces for color-texture retrieval, in: International Conference on Image Analysis and Recognition, pp. 354–363.
[41] T. Lindeberg, Scale-Space Theory in Computer Vision, Kluwer Academic Publishers, 1994.
[42] R. Gonzalez, R. Woods, Digital Image Processing, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1992.
[43] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000) 888–905.
[44] K. Plataniotis, A. Venetsanopoulos, Color image processing and applications, Springer-Verlag New York, Inc., New York, NY, USA, pp. 237–277.
[45] J. Serra, Image analysis and mathematical morphology II. Theoretical Advances, volume 2, Academic Press, 1988.
[46] J. Malik, P. Perona, Preattentive texture discrimination with early vision mechanisms, Journal of the Optical Society of America 7 (1990) 923–932.
[47] B. Berlin, P. Kay, Basic Color Terms: Their Universality and Evolution, University of California Press, Berkeley, CA, 1969.
[48] R. Benavente, M. Vanrell, R. Baldrich, Estimation of fuzzy sets for computational colour categorization, Color Research and Application 29 (2004) 342–353.
[49] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen, S. Huovinen, Outex - new framework for empirical evaluation of texture analysis algorithms, in: Proc. 16th International Conference on Pattern Recognition, volume 1, pp. 701–706.
[50] Y. Rubner, C. Tomasi, L. Guibas, The earth mover's distance as a metric for image retrieval, International Journal of Computer Vision 40 (2000) 99–121.
[51] G. Finlayson, G. Tian, Investigating colour texture similarity, in: Infotech Oulu Workshop on Texture Analysis in Machine Vision, pp. 127–136.
[52] J. Smith, Image retrieval evaluation, in: Proc. IEEE Workshop on Content - Based Access of Image and Video Libraries, IEEE Computer Society, Los Alamitos, CA, USA, 1998, pp. 112–113.
[53] B. Manjunath, P. Salembier, T. Sikora, Introduction to MPEG-7, John Wiley & Sons, 2003.
[54] T. Kohonen, Self-Organizing maps, Springer-Verlag, 1997.
[55] J. Kruskal, M. Wish, Multidimensional Scaling, Sage Publications, Inc., 1978.