EURASIP Journal on Applied Signal Processing 2004:6, 886–901
© 2004 Hindawi Publishing Corporation

Region-Based Image Retrieval Using an Object Ontology and Relevance Feedback

Vasileios Mezaris

Information Processing Laboratory, Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

Centre for Research and Technology Hellas (CERTH), Informatics and Telematics Institute (ITI), 57001 Thessaloniki, Greece
Email: bmezaris@iti.gr

Ioannis Kompatsiaris

Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

Centre for Research and Technology Hellas (CERTH), Informatics and Telematics Institute (ITI), 57001 Thessaloniki, Greece
Email: ikom@iti.gr

Michael G. Strintzis

Information Processing Laboratory, Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

Centre for Research and Technology Hellas (CERTH), Informatics and Telematics Institute (ITI), 57001 Thessaloniki, Greece
Email: strintzi@eng.auth.gr

Received 31 January 2003; Revised 3 September 2003

An image retrieval methodology suited for search in large collections of heterogeneous images is presented. The proposed approach employs a fully unsupervised segmentation algorithm to divide images into regions and endow the indexing and retrieval system with content-based functionalities. Low-level descriptors for the color, position, size, and shape of each region are subsequently extracted. These arithmetic descriptors are automatically associated with appropriate qualitative intermediate-level descriptors, which form a simple vocabulary termed *object ontology*. The object ontology is used to allow the qualitative definition of the high-level concepts the user queries for (*semantic objects*, each represented by a *keyword*) and their relations in a human-centered fashion. When querying for a specific semantic object (or objects), the intermediate-level descriptor values associated with both the semantic object and all image regions in the collection are initially compared, resulting in the rejection of most image regions as irrelevant. Following that, a relevance feedback mechanism, based on support vector machines and using the low-level descriptors, is invoked to rank the remaining potentially relevant image regions and produce the final query results. Experimental results and comparisons demonstrate, in practice, the effectiveness of our approach.

Keywords and phrases: image retrieval, image databases, image segmentation, ontology, relevance feedback, support vector machines.

1. INTRODUCTION

In recent years, the accelerated growth of digital media collections and in particular still image collections, both proprietary and on the web, has established the need for the development of human-centered tools for the efficient access and retrieval of visual information. As the amount of information available in the form of still images continuously increases, the necessity of efficient methods for the retrieval of the visual information becomes evident [1]. Additionally, the continuously increasing number of people with access to such image collections further dictates that more emphasis must be put on attributes such as the user-friendliness and flexibility of any image-retrieval scheme. These facts, along with the diversity of available image collections, varying from *restricted*, for example, medical image databases and satellite photo collections, to general purpose collections, which contain heterogeneous images, and the diversity of requirements regarding the amount of knowledge about the images that should be used for indexing, have led to the development of a wide range of solutions [2].

The very first attempts for image retrieval were based on exploiting existing image captions to classify images to predetermined classes or to create a restricted vocabulary [3].

Although relatively simple and computationally efficient, this approach has several restrictions mainly deriving from the use of a restricted vocabulary that neither allows for unanticipated queries nor can be extended without reevaluating the possible connection between each image in the database and each new addition to the vocabulary. Additionally, such keyword-based approaches assume either the preexistence of textual image annotations (e.g., captions) or that annotation, using the predetermined vocabulary, is performed manually. In the latter case, inconsistency of the keyword assignments among different indexers can also hamper performance. Recently, a methodology for computer-assisted annotation of image collections was presented [4].

To overcome the limitations of the keyword-based approach, the use of the image visual contents has been proposed. This category of approaches utilizes the visual contents by extracting low-level indexing features for each image or image segment (region). Then, relevant images are retrieved by comparing the low-level features of each item in the database with those of a user-supplied sketch [5], or, more often, a key image that is either selected from a restricted image set or is supplied by the user (*query by example*). One of the first attempts to realize this scheme is the *query by image content* system [6, 7]. Newer contributions to query by example (QbE) include systems such as *NeTra* [8, 9], *Mars* [10], *Photobook* [11], *VisualSEEK* [12], and *Istorama* [13]. They all employ the general framework of QbE, demonstrating the use of various indexing feature sets either in the *image* or in the *region* domain.

A recent addition to this group, Berkeley's *Blobworld* [14, 15], proposes segmentation using the expectation-maximization algorithm and clearly demonstrates the improvement in query results attained by querying using region-based indexing features rather than global image properties, under the QbE scheme. Other works on segmentation, that can be of use in content-based retrieval, include segmentation by anisotropic diffusion [16], the RSST algorithm [17], the watershed transformation [18], the normalized cut [19], and the mean shift approach [20]. While such segmentation algorithms can endow an indexing and retrieval system with extensive content-based functionalities, these are limited by the main drawback of QbE approaches, that is, the need for the availability of an appropriate key image in order to start a query. Occasionally, satisfying this condition is not feasible, particularly for image classes that are under-represented in the database.

Hybrid methods exploiting both keywords and the image visual contents have also been proposed [21, 22, 23]. In [21], the use of *probabilistic multimedia objects* (*multijects*) is proposed; these are built using hidden Markov models and necessary training data. Significant work was recently presented on unifying keywords and visual contents in image retrieval. The method of [23] performs semantic grouping of keywords based on user relevance feedback to effectively address issues such as word similarity and allow for more efficient queries; nevertheless, it still relies on preexisting or manually added textual annotations. In well-structured specific domain applications (e.g., sports and news broadcasting) domain-specific features that facilitate the modelling of higher-level semantics can be extracted [24, 25]. A priori knowledge representation models are used as a knowledge base that assists semantic-based classification and clustering. In [26], *semantic entities*, in the context of the MPEG-7 standard, are used for knowledge-assisted video analysis and object detection, thus allowing for semantic-level indexing. However, the need for accurate definition of semantic entities using low-level features restricts this kind of approaches to domain-specific applications and prohibits nonexperts from defining new semantic entities.

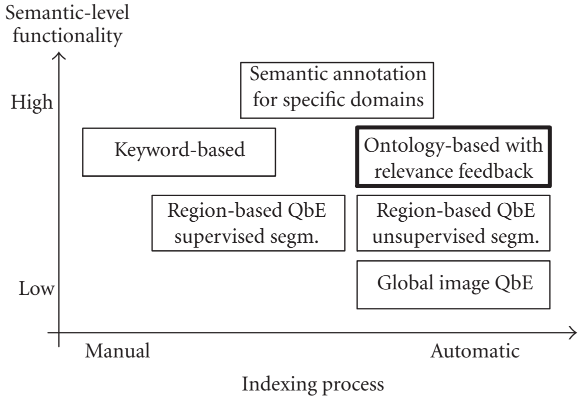

Semantic-level functionality
High
Low
Semantic annotation for specific domains
Keyword-based
Ontology-based with relevance feedback
Region-based QbE supervised segm.
Region-based QbE unsupervised segm.
Global image QbE
Manual
Automatic
Indexing process

Figure 1: Overview of image retrieval techniques. Techniques exploiting preexisting textual information (e.g., captions) associated with the images would lie in the same location on the diagram as the proposed approach, but are limited to applications where such a priori knowledge is available.

This paper attempts to address the problem of retrieval in generic image collections, where no possibility of structuring a domain-specific knowledge base exists, without imposing restrictions such as the availability of key images or image captions. The adopted region-based approach employs still image segmentation tools that enable the time-efficient and unsupervised analysis of still images to regions, thus allowing the "content-based" access and manipulation of visual data via the extraction of low-level indexing features for each region. To take further advantage of the human-friendly aspects of the region-based approach, the low-level indexing features for the spatial regions can be associated with higher-level concepts that humans are more familiar with. This is achieved with the use of an *ontology* and a relevance feedback mechanism [27, 28]. Ontologies [29, 30, 31] define a formal language for the structuring and storage of the high-level features, facilitate the mapping of low-level to high-level features, and allow the definition of relationships between pieces of multimedia information; their potential applications range from text retrieval [32] to facial expression recognition [33]. The resulting image indexing and retrieval scheme provides flexibility in defining the desired semantic object/keyword and bridges the gap between keyword-based approaches and QbE approaches (Figure 1).

The paper is organized as follows. The employed image segmentation algorithm is presented in Section 2. Section 3 presents in detail the components of the retrieval scheme. Section 4 contains an experimental evaluation and comparisons of the developed methods, and finally, conclusions are drawn in Section 5.

2. COLOR IMAGE SEGMENTATION

2.1. *Segmentation algorithm overview*

A region-based approach to image retrieval has been adopted; thus, the process of inserting an image into the database starts by applying a color image segmentation algorithm to it, so as to break it down to a number of regions. The segmentation algorithm employed for the analysis of images to regions is based on a variant of the K-means with connectivity constraint algorithm (KMCC), a member of the popular K-means family [34]. The KMCC algorithm classifies the pixels into regions s_k, $k = 1, \ldots, K$, taking into account not only the intensity of each pixel but also its position, thus producing connected regions rather than sets of chromatically similar pixels. In the past, KMCC has been successfully used for model-based image sequence coding [35] and content-based watermarking [36]. The variant used for the purpose of still image segmentation [37] additionally uses texture features in combination with the intensity and position features.

The overall segmentation algorithm consists of the following stages.

Stage 1. Extraction of the intensity and texture feature vectors corresponding to each pixel. These will be used along with the spatial features in the following stages.

Stage 2. Estimation of the initial number of regions and their spatial, intensity, and texture centers, using an initial clustering procedure. These values are to be used by the KMCC algorithm.

Stage 3. Conditional filtering using a moving average filter.

Stage 4. Final classification of the pixels, using the KMCC algorithm.

The result of the application of the segmentation algorithm to a color image is a segmentation mask M, that is, a gray-scale image comprising the spatial regions formed by the segmentation algorithm, $M = \{s_1, s_2, \ldots, s_K\}$, in which different gray values, $1, 2, \ldots, K$, correspond to different regions, $M(\mathbf{p} \in s_k) = k$, where $\mathbf{p} = [p_x \;\; p_y]^T$, $p_x = 1, \ldots, x_{\max}$, $p_y = 1, \ldots, y_{\max}$ are the image pixels and $x_{\max}$, $y_{\max}$ are the image dimensions. This mask is used for extracting the region low-level indexing features described in Section 3.1.

2.2. *Color and texture features*

For every pixel $\mathbf{p}$, a color feature vector and a texture feature vector are calculated. The three intensity components of the CIE $L^*a^*b^*$ color space are used as intensity features, $\mathbf{I}(\mathbf{p}) = [I_L(\mathbf{p}) \;\; I_a(\mathbf{p}) \;\; I_b(\mathbf{p})]^T$, since it has been shown that $L^*a^*b^*$ is more suitable for segmentation than the widely used RGB color space, due to its being approximately perceptually uniform [38].

In order to detect and characterize texture properties in the neighborhood of each pixel, the discrete wavelet frames (DWF) [39] decomposition of two levels is used. The employed filter bank is based on the low-pass Haar filter $H(z) = (1/2)(1 + z^{-1})$, which satisfies the low-pass condition $H(z)|_{z=1} = 1$. The complementary high-pass filter $G(z)$ is defined by $G(z) = zH(-z^{-1})$. The filters of the filter bank are then generated by the prototypes $H(z)$, $G(z)$, as described in [39]. Despite its simplicity, the above filter bank has been demonstrated to perform surprisingly well for texture segmentation, while featuring reduced computational complexity. The texture feature vector $\mathbf{T}(\mathbf{p})$ is then made of the standard deviations of all detail components, calculated in a square neighborhood Φ of pixel $\mathbf{p}$.

2.3. *Initial clustering*

An initial estimation of the number of regions in the image and their spatial, intensity, and texture centers is required for the initialization of the KMCC algorithm. In order to compute these initial values, the image is broken down to square, nonoverlapping blocks of dimension $f \times f$. In this way, a reduced image composed of a total of L blocks, b_l, $l = 1, \ldots, L$, is created. A color feature vector $\mathbf{I}^b(b_l) = [I_L^b(b_l) \;\; I_a^b(b_l) \;\; I_b^b(b_l)]^T$ and a texture feature vector $\mathbf{T}^b(b_l)$ are then assigned to each block; their values are estimated as the averages of the corresponding features for all pixels belonging to the block. The distance between two blocks is defined as follows:

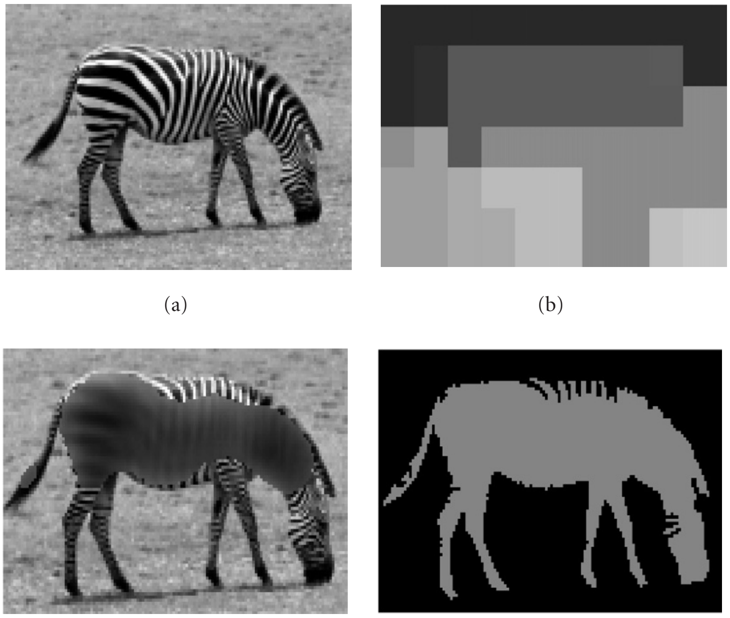
$$D^b(b_l, b_n) = \|\mathbf{I}^b(b_l) - \mathbf{I}^b(b_n)\| + \lambda_1 \|\mathbf{T}^b(b_l) - \mathbf{T}^b(b_n)\|, \qquad (1)$$

where $\|\mathbf{I}^b(b_l) - \mathbf{I}^b(b_n)\|$, $\|\mathbf{T}^b(b_l) - \mathbf{T}^b(b_n)\|$ are the Euclidean distances between the block feature vectors. In our experiments, $\lambda_1 = 1$, since experimentation showed that using a different weight λ_1 for the texture difference would result in erroneous segmentation of textured images if $\lambda_1 \ll 1$, respectively, nontextured images if $\lambda_1 \gg 1$. As shown in the experimental results section, the value $\lambda_1 = 1$ is appropriate for a variety of textured and nontextured images.

The number of regions of the image is initially estimated by applying a variant of the maximin algorithm [40] to this set of blocks. The distance C between the first two centers identified by the maximin algorithm is indicative of the intensity and texture contrast of the particular image. Subsequently, a simple K-means algorithm is applied to the set of blocks, using the information produced by the maximin algorithm for its initialization. Upon convergence, a recursive four-connectivity component labelling algorithm [41] is applied so that a total of K' connected regions s_k, $k = 1, \ldots, K'$, are identified. Their intensity, texture, and spatial centers, $\mathbf{I}^s(s_k)$, $\mathbf{T}^s(s_k)$, and $\mathbf{S}(s_k) = [S_x(s_k) \;\; S_y(s_k)]^T$, $k = 1, \ldots, K'$, are calculated as follows:

$$\mathbf{I}^s(s_k) = \frac{1}{A_k} \sum_{\mathbf{p} \in s_k} \mathbf{I}(\mathbf{p}), \qquad \mathbf{T}^s(s_k) = \frac{1}{A_k} \sum_{\mathbf{p} \in s_k} \mathbf{T}(\mathbf{p}),$$
$$\mathbf{S}(s_k) = \frac{1}{A_k} \sum_{\mathbf{p} \in s_k} \mathbf{p}, \qquad (2)$$

where A_k is the number of pixels belonging to region s_k: $s_k = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_{A_k}\}$.



(a)

(b)

(c)

(d)

Figure 2: Segmentation process starting from (a) the original image, (b) initial clustering and (c) conditional filtering are performed and (d) final results are produced.

2.4. Conditional filtering

Images may contain parts in which intensity fluctuations are particularly pronounced, even when all pixels in these parts of the image belong to a single object (Figure 2). In order to facilitate the grouping of all these pixels in a single region based on their texture similarity, a moving average filter is employed. The decision of whether the filter should be applied to a particular pixel $\mathbf{p}$ or not is made by evaluating the norm of the texture feature vector $\mathbf{T}(\mathbf{p})$ (Section 2.2); the filter is not applied if that norm is below a threshold τ. The output of the conditional filtering module can thus be expressed as

$$\mathbf{J}(\mathbf{p}) = \begin{cases} \mathbf{I}(\mathbf{p}), & \text{if } \|\mathbf{T}(\mathbf{p})\| < \tau, \\ \dfrac{1}{f^2} \sum \mathbf{I}(\mathbf{p}), & \text{if } \|\mathbf{T}(\mathbf{p})\| \geq \tau. \end{cases} \tag{3}$$

Correspondingly, region intensity centers calculated similarly to (2) using the filtered intensities $\mathbf{J}(\mathbf{p})$ instead of $\mathbf{I}(\mathbf{p})$ are symbolized by $\mathbf{J}^s(s_k)$.

An appropriate value of threshold τ was experimentally found to be

$$\tau = \max \{0.65 \cdot T_{\max}, 14\}, \tag{4}$$

where $T_{\max}$ is the maximum value of the norm $\|\mathbf{T}(\mathbf{p})\|$ in the image. The term $0.65 \cdot T_{\max}$ in the threshold definition serves to prevent the filter from being applied outside the borders of textured objects, so that their boundaries are not corrupted. The constant bound 14, on the other hand, is used to prevent the filtering of images composed of chromatically uniform objects. In such images, the value of $T_{\max}$ is expected to be relatively small and would correspond to pixels on edges between objects, where filtering is obviously undesirable.

2.5. The K-means with connectivity constraint algorithm

The KMCC algorithm applied to the pixels of the image consists of the following steps.

Step 1. The region number and the region centers are initialized using the output of the initial clustering procedure described in Section 2.3.

Step 2. For every pixel $\mathbf{p}$, the distance between $\mathbf{p}$ and all region centers is calculated. The pixel is then assigned to the region for which the distance is minimized. A generalized distance of a pixel $\mathbf{p}$ from a region s_k is defined as follows:

$$\begin{aligned} D(\mathbf{p}, s_k) = {} & \|\mathbf{J}(\mathbf{p}) - \mathbf{J}^s(s_k)\| + \lambda_1 \|\mathbf{T}(\mathbf{p}) - \mathbf{T}^s(s_k)\| \\ & + \lambda_2 \frac{\bar{A}}{A_k} \|\mathbf{p} - \mathbf{S}(s_k)\|, \end{aligned} \tag{5}$$

where $\|\mathbf{J}(\mathbf{p}) - \mathbf{J}^s(s_k)\|$, $\|\mathbf{T}(\mathbf{p}) - \mathbf{T}^s(s_k)\|$, and $\|\mathbf{p} - \mathbf{S}(s_k)\|$ are the Euclidean distances between the pixel feature vectors and the corresponding region centers, the pixel number A_k of region s_k is a measure of the area of region s_k, and $\bar{A}$ is the average area of all regions, $\bar{A} = (1/K) \sum_{k=1}^{K} A_k$. The regularization parameter λ_2 is defined as $\lambda_2 = 0.4 \cdot C / \sqrt{x_{\max}^2 + y_{\max}^2}$, while the choice of the parameter λ_1 has been discussed in Section 2.3.

In (5), the normalization of the spatial distance, $\|\mathbf{p} - \mathbf{S}(s_k)\|$ by division by the area of each region $A_k/\bar{A}$, is necessary in order to encourage the creation of large connected regions, otherwise, pixels would tend to be assigned to smaller rather than larger regions due to greater spatial proximity to their centers. The regularization parameter λ_2 is used to ensure that a pixel is assigned to a region primarily due to their similarity in intensity and texture characteristics, even in low-contrast images, where intensity and texture differences are small compared to spatial distances.

Step 3. The connectivity of the formed regions is evaluated. Those which are not connected are broken down to the minimum number of connected regions using a recursive four-connectivity component labelling algorithm [41].

Step 4. Region centers are recalculated (2). Regions whose area size lies below a threshold ξ are dropped. In our experiments, the threshold ξ was equal to 0.5% of the total image area. The number of regions K is then recalculated, taking into account only the remaining regions.

Step 5. Two regions are merged if they are neighbors and if their intensity and texture distance is not greater than an appropriate merging threshold:

$$\begin{aligned} &D^s(s_{k_1}, s_{k_2}) \\ &\quad = \|\mathbf{J}^s(s_{k_1}) - \mathbf{J}^s(s_{k_2})\| + \lambda_1 \|\mathbf{T}^s(s_{k_1}) - \mathbf{T}^s(s_{k_2})\| \le \mu. \end{aligned} \tag{6}$$

The threshold μ is image-specific, defined in our experiments by

$$\mu = \begin{cases} 7.5, & \text{if } C < 25, \\ 15, & \text{if } C > 75, \\ 10, & \text{otherwise,} \end{cases} \tag{7}$$

where C is an approximation of the intensity and texture contrast of the particular image, as defined in Section 2.3

Step 6. Region number K and region centers are reevaluated.

Step 7. If the region number K is equal to the one calculated in Step 6 of the previous iteration and the difference between the new centers and those in Step 6 of the previous iteration is below the corresponding threshold for all centers, then stop, else go to Step 2. If index "old" characterizes the region number and region centers calculated in Step 6 of the previous iteration, the convergence condition can be expressed as $K = K^{\text{old}}$ and

$$\begin{gathered} \left\|\mathbf{J}^s(s_k) - \mathbf{J}^s\left(s_k^{\text{old}}\right)\right\| \le c_I, \qquad \left\|\mathbf{T}^s(s_k) - \mathbf{T}^s\left(s_k^{\text{old}}\right)\right\| \le c_T, \\ \left\|\mathbf{S}(s_k) - \mathbf{S}\left(s_k^{\text{old}}\right)\right\| \le c_S, \end{gathered} \tag{8}$$

for $k = 1, \ldots, K$. Since there is no certainty that the KMCC algorithm will converge for any given image, the maximum allowed number of iterations was chosen to be 20; if this is exceeded, the method proceeds as though the KMCC algorithm had converged.

3. REGION-BASED RETRIEVAL SCHEME

3.1. Low-level indexing descriptors

As soon as the segmentation mask is produced, a set of descriptors that will be useful in querying the database are calculated for each region. These region descriptors compactly characterize each region's color, position, and shape. All descriptors are normalized so as to range from 0 to 1.

The color and position descriptors of a region are the normalized intensity and spatial centers of the region. In particular, the *color descriptors* of region s_k, F_1, F_2, F_3, corresponding to the L, a, b components, are defined as follows:

$$\begin{aligned} F_1 &= \frac{1}{100 \cdot A_k} \sum_{\mathbf{p} \in s_k} I_L(\mathbf{p}), \\ F_2 &= \frac{(1/A_k) \sum_{\mathbf{p} \in s_k} I_a(\mathbf{p}) + 80}{200}, \\ F_3 &= \frac{(1/A_k) \sum_{\mathbf{p} \in s_k} I_b(\mathbf{p}) + 80}{200}, \end{aligned} \tag{9}$$

where A_k is the number of pixels belonging to region s_k. Similarly, the *position descriptors* F_4, F_5 are defined as

$$F_4 = \frac{1}{A_k \cdot x_{\max}} \sum_{\mathbf{p} \in s_k} p_x, \qquad F_5 = \frac{1}{A_k \cdot y_{\max}} \sum_{\mathbf{p} \in s_k} p_y. \tag{10}$$

Although quantized color histograms are considered to provide a more detailed description of a region's colors than intensity centers, they were not chosen as color descriptors, since this would significantly increase the dimensionality of the feature space, thus increasing the time complexity of the query execution.

The *shape descriptors* F_6, F_7 of a region are its normalized area and eccentricity. We chose not to take into account the orientation of regions, since orientation is hardly characteristic of an object. The normalized area F_6 is expressed by the number of pixels A_k that belong to region s_k, divided by the total number of pixels of the image:

$$F_6 = \frac{A_k}{x_{\max} \cdot y_{\max}}. \tag{11}$$

The eccentricity is calculated using the covariance or scatter matrix $\mathbf{C}_k$ of the region. This is defined as

$$\mathbf{C}_k = \frac{1}{A_k} \sum_{\mathbf{p} \in s_k} (\mathbf{p} - \mathbf{S}(s_k))(\mathbf{p} - \mathbf{S}(s_k))^T, \tag{12}$$

where $\mathbf{S}(s_k) = [S_x(s_k) \;\; S_y(s_k)]^T$ is the region spatial center. Let ρ_i, $\mathbf{u}_i$, $i = 1, 2$, be its eigenvalues and eigenvectors, $\mathbf{C}_k \mathbf{u}_i = \rho_i \mathbf{u}_i$ with $\mathbf{u}_i^T \mathbf{u}_i = 1$, $\mathbf{u}_i^T \mathbf{u}_j = 0$, $i \ne j$, and $\rho_1 \ge \rho_2$. According to the principal component analysis (PCA), the principal eigenvector $\mathbf{u}_1$ defines the orientation of the region and $\mathbf{u}_2$ is perpendicular to $\mathbf{u}_1$. The two eigenvalues provide an approximate measure of the two dominant directions of

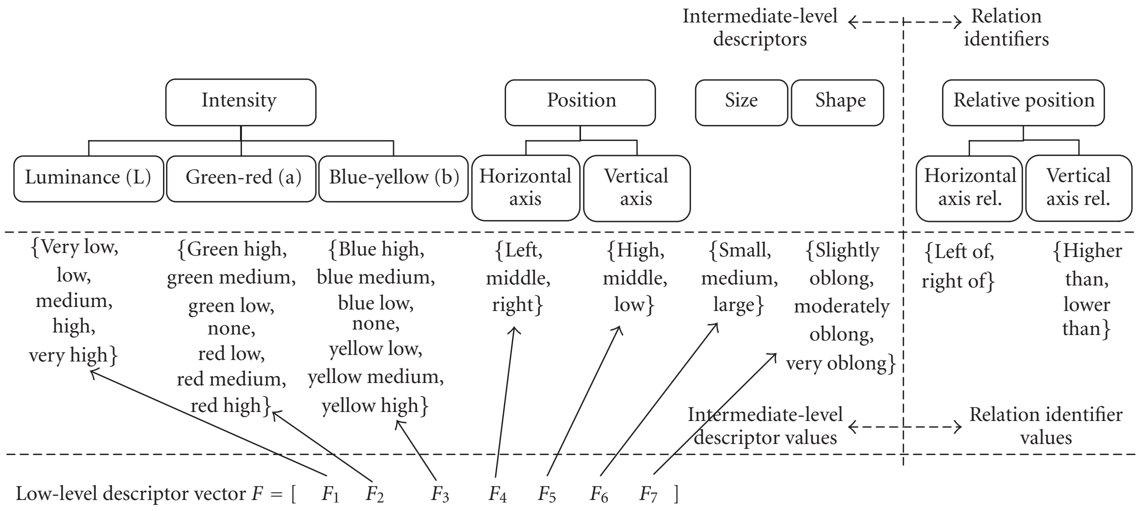

Intermediate-level descriptors
Relation identifiers
Intensity
Position
Size
Shape
Relative position
Luminance (L)
Green-red (a)
Blue-yellow (b)
Horizontal axis
Vertical axis
Horizontal axis rel.
Vertical axis rel.
{Very low, low, medium, high, very high}
{Green high, green medium, green low, none, red low, red medium, red high}
{Blue high, blue medium, blue low, none, yellow low, yellow medium, yellow high}
{Left, middle, right}
{High, middle, low}
{Small, medium, large}
{Slightly oblong, moderately oblong, very oblong}
{Left of, right of}
{Higher than, lower than}
Intermediate-level descriptor values
Relation identifier values
Low-level descriptor vector F = [F_1 F_2 F_3 F_4 F_5 F_6 F_7]

FIGURE 3: Object ontology: the intermediate-level descriptors are the elements of set $\mathcal{D}$ whereas the relation identifiers are the elements of set $\mathcal{R}$.

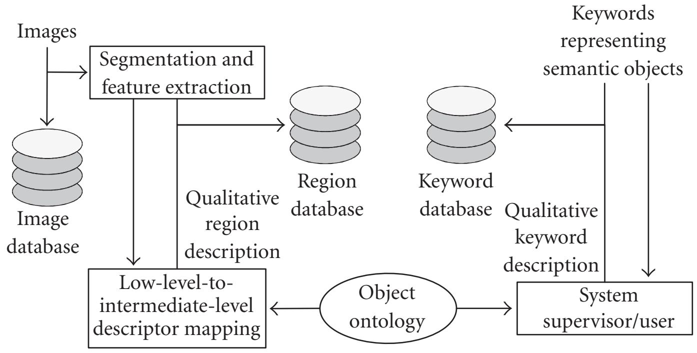

Images
Segmentation and feature extraction
Image database
Region database
Keyword database
Keywords representing semantic objects
Qualitative region description
Qualitative keyword description
Low-level-to-intermediate-level descriptor mapping
Object ontology
System supervisor/user

FIGURE 4: Indexing system overview: low-level and intermediate-level descriptor values for the regions are stored in the region database; intermediate-level descriptor values for the user-defined keywords (semantic objects) are stored in the keyword database.

the shape. Using these quantities, an approximation of the eccentricity ε_k of the region is calculated as follows:

$$\varepsilon_k = 1 - \frac{\rho_1}{\rho_2}. \tag{13}$$

The normalized eccentricity descriptor F_7 is then defined as $F_7 = e^{\varepsilon_k}$.

The seven region descriptors defined above form a region descriptor vector $\mathbf{F}$:

$$\mathbf{F} = \left[F_1 \quad \cdots \quad F_7 \right]. \tag{14}$$

This region descriptor vector will be used in the sequel both for assigning intermediate-level qualitative descriptors to the region and as an input to the relevance feedback mechanism. In both cases, the existence of these low-level descriptors is not apparent to the end user.

3.2. *Object ontology*

In this work, an ontology is employed to allow the user to query an image collection using semantically meaningful concepts (semantic objects), as in [42]. As opposed to [42], though, no manual annotation of images is performed. Instead, a simple *object ontology* is used to enable the user to describe semantic objects, like "tiger," and relations between semantic objects, using a set of *intermediate-level descriptors* and *relation identifiers* (Figure 3). The architecture of this indexing scheme is illustrated in Figure 4. The simplicity of the employed object ontology serves the purpose of it being applicable to generic image collections without requiring the correspondence between image regions and relevant identifiers be defined manually. The object ontology can be expanded so as to include additional descriptors and relation identifiers corresponding either to low-level region properties (e.g., texture) or to higher-level semantics which, in domain-specific applications, could be inferred either from

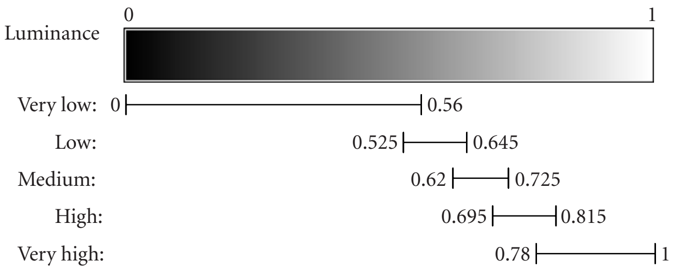


Figure 5: Correspondence of low-level and intermediate-level descriptor values for the luminance descriptor.

the visual information itself or from associated information (e.g., text), should there be any. Similar to [43], an ontology is defined as follows.

Definition 1. An *object ontology* is a structure (Figure 3)

$$\mathcal{O} := \left(\mathcal{D}, \leq_{\mathcal{D}}, \mathcal{R}, \sigma, \leq_{\mathcal{R}}\right) \tag{15}$$

consisting of the following. (i) Two disjoint sets $\mathcal{D}$ and $\mathcal{R}$ whose elements d and r are called, respectively, intermediate-level descriptors (e.g., intensity, position, etc.) and relation identifiers (e.g., relative position). To simplify the terminology, relation identifiers will often be called *relations* in the sequel. The elements of set $\mathcal{D}$ are often called *concept identifiers* or *concepts* in the literature. (ii) A partial order $\leq_{\mathcal{D}}$ on $\mathcal{D}$ is called concept hierarchy or taxonomy (e.g., luminance is a subconcept of intensity). (iii) A function $\sigma : \mathcal{R} \rightarrow \mathcal{D}^+$ is called *signature*; $\sigma(r) = (\sigma_{1,r}, \sigma_{2,r}, \ldots, \sigma_{\Sigma,r})$, $\sigma_{i,r} \in \mathcal{D}$ and $|\sigma(r)| \equiv \Sigma$ denotes the number of elements of $\mathcal{D}$ on which $\sigma(r)$ depends. (iv) A partial order $\leq_{\mathcal{R}}$ on $\mathcal{R}$ is called relation hierarchy, where $r_1 \leq_{\mathcal{R}} r_2$ implies $|\sigma(r_1)| = |\sigma(r_2)|$ and $\sigma_{i,r_1} \leq_{\mathcal{D}} \sigma_{i,r_2}$ for each $1 \leq i \leq |\sigma(r_1)|$.

For example, the signature of relation r relative position, is by definition $\sigma(r) =$ ("position," "position"), indicating that it relates a position to a position, where $|\sigma(r)| = 2$ denotes that r involves two elements of set $\mathcal{D}$. Both the intermediate-level "position" descriptor values and the underlying low-level descriptor values can be employed by the relative position relation.

In Figure 3, the possible intermediate-level descriptors and descriptor values are shown. Each value of these intermediate-level descriptors is mapped to an appropriate range of values of the corresponding low-level, arithmetic descriptor. The various value ranges for every low-level descriptor are chosen so that the resulting intervals are equally populated. This is pursued so as to prevent an intermediate-level descriptor value from being associated with a majority of image regions in the database, because this would render it useless in restricting a query to the potentially most relevant ones. Overlapping, up to a point, of adjacent value ranges is used to introduce a degree of fuzziness to the descriptors; for example, both "low luminance" and "medium luminance" values may be used to describe a single region.

Let $d_{q,z}$ be the qth descriptor value (e.g., low luminance) of intermediate-level descriptor d_z (e.g., luminance) and let $R_{q,z} = [L_{q,z}, H_{q,z}]$ be the range of values of the corresponding arithmetic descriptor F_m (14). Given the probability density function pdf(F_m), the overlapping factor V expressing the degree of overlapping of adjacent value ranges, and given that value ranges should be equally populated, lower and upper bounds $L_{q,z}$, $H_{q,z}$ can be easily calculated as follows:

$$\begin{gathered}
L_{1,z} = L_m, \quad \int_{L_{q-1,z}}^{L_{q,z}} \text{pdf}\left(F_m\right) dF_m = \frac{1-V}{Q_z - V \cdot \left(Q_z - 1\right)}, \\
\int_{L_{1,z}}^{H_{1,z}} \text{pdf}\left(F_m\right) dF_m = \frac{1}{Q_z - V \cdot \left(Q_z - 1\right)}, \\
\int_{H_{q-1,z}}^{H_{q,z}} \text{pdf}\left(F_m\right) dF_m = \frac{1-V}{Q_z - V \cdot \left(Q_z - 1\right)},
\end{gathered} \tag{16}$$

where $q = 2, \ldots, Q_z$, Q_z is the number of descriptor values defined for descriptor d_z (e.g., for luminance, $Q_z = 5$), and L_m is the lower bound of the values of F_m. Note that for descriptors "green-red" and "blue-yellow," the above process is performed twice: once for each of the two complementary colors described by each descriptor, taking into account each time the appropriate range of values of the corresponding low-level descriptor. Lower and upper bounds for value "none" of the descriptor green-red are chosen so as to associate with this value a fraction V of the population of descriptor value "green low" and a fraction V of the population of descriptor value "red low;" bounds for value none of descriptor blue-yellow are defined accordingly. The overlapping factor V is defined as $V = 0.25$ in our experiments. The boundaries calculated by the above method for the luminance descriptor, using the image database defined in Section 4, are presented in Figure 5.

3.3. Query process

A query is formulated using the object ontology to provide a qualitative definition of the sought object or objects (using the intermediate-level descriptors) and the relations between them. Definitions previously imported to the system by the same or other users can also be employed, as discussed in the sequel. As soon as a query is formulated, the

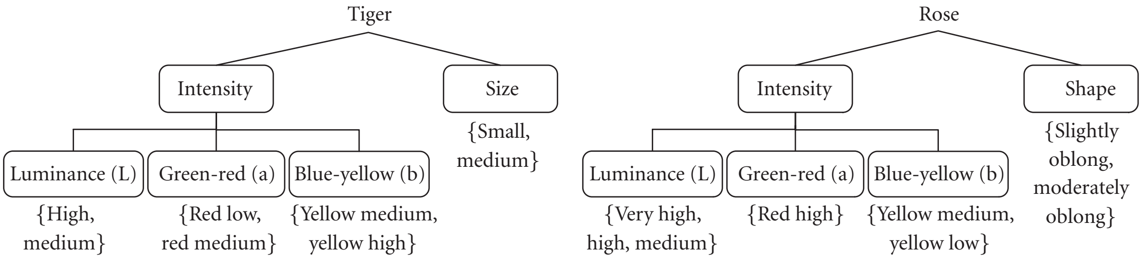


Figure 6: Exemplary keyword definitions using the object ontology.

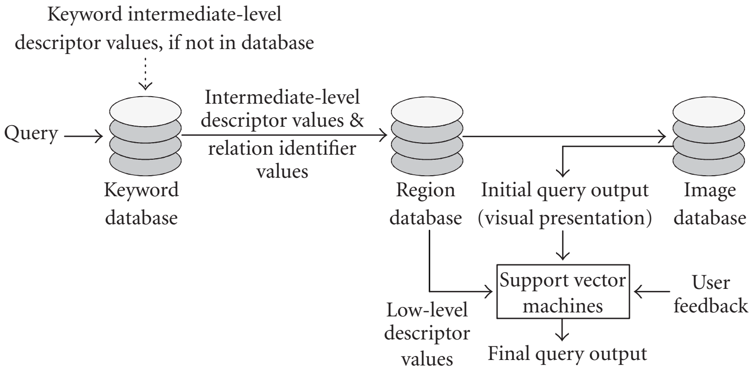

Keyword intermediate-level descriptor values, if not in database
Query
Keyword database
Intermediate-level descriptor values & relation identifier values
Region database
Initial query output (visual presentation)
Image database
Support vector machines
User feedback
Low-level descriptor values
Final query output

Figure 7: Query process overview.

intermediate-level descriptor values associated with each desired object/keyword are compared to those of each image region contained in the database. Descriptors for which no values have been associated with the desired object (e.g., “shape” for object “tiger,” defined in Figure 6) are ignored; for each remaining descriptor, regions not sharing at least one descriptor value with those assigned to the desired object are deemed irrelevant (e.g., a region with size “large” is not a potentially relevant region for a “tiger” query, as opposed to a region assigned both “large” and “medium” values for its “size” descriptor). In the case of dual-keyword queries, the above process is performed for each keyword separately and only images containing at least two distinct potentially relevant regions, one for each keyword, are returned. If desired spatial relations between the queried objects have been defined, compliance with them is checked using the corresponding region intermediate-level and low-level descriptors, to further reduce the number of potentially relevant images returned to the user.

After narrowing down the search to a set of potentially relevant image regions, relevance feedback is employed to produce a quantitative evaluation of the degree of relevance of each region. The employed mechanism is based on a method proposed in [44], where it is used for image retrieval using global image properties under the QbE scheme. This combines support vector machines (SVMs) [45, 46] with a constrained similarity measure (CSM) [44]. SVMs employ the user-supplied feedback (training samples) to learn the boundary separating the two classes (positive and negative samples, respectively). Each sample (in our case, image region) is represented by its low-level descriptor vector **F** (Section 3.1). Following the boundary estimation, the CSM is employed to provide a ranking; in [44], the CSM employs the Euclidean distance from the key image used for initiating the query for images inside the boundary (images classified as relevant) and the distance from the boundary for those classified as irrelevant. Under the proposed scheme, no key image is used for query initiation; the CSM is therefore modified so as to assign to each image region classified as relevant the minimum of the Euclidean distances between it and all positive training samples (i.e., image regions marked as relevant by the user during relevance feedback). The query procedure is graphically illustrated in Figure 7.

The relevance feedback process can be repeated as many times as necessary, each time using all the previously supplied training samples. Furthermore, it is possible to store the parameters of the trained SVM and the corresponding training set for every keyword that has already been used in a query at least once. This endows the system with the capability to respond to anticipated queries without initially requiring any feedback; in a multiuser (e.g., web-based) environment, it additionally enables different users to share knowledge either in the form of semantic object descriptions or in the form of results retrieved from the database. In either case, further refinement of retrieval results is possible by additional rounds of relevance feedback.

Table 1: Numerical evaluation of segmentation results of Figures 8 and 9.

Classes	Image 1		Image 2		Image 3	
	Blobworld	Proposed	Blobworld	Proposed	Blobworld	Proposed
Eagle	163.311871	**44.238528**	16.513599	**7.145284**	11.664597	**2.346432**
Tiger	90.405821	**12.104017**	**47.266126**	57.582892	86.336678	**12.979979**
Car	133.295750	**54.643714**	54.580259	**27.884972**	122.057933	**4.730332**
Rose	37.524702	**2.853145**	184.257505	**1.341963**	**22.743732**	53.501481
Horse	65.303681	**17.350378**	22.099393	**12.115678**	233.303729	**120.862361**

4. EXPERIMENTAL RESULTS

The proposed algorithms were tested on a collection of 5000 images from the Corel gallery.[1] Application of the segmentation algorithm of Section 2 to these images resulted in the creation of a database containing 34433 regions, each represented by a low-level descriptor vector, as discussed in Section 3.1. The segmentation and low-level feature extraction are required on the average 27.15 seconds and 0.011 seconds, respectively, on a 2 GHz Pentium IV PC. The proposed algorithm was compared with the Blobworld segmentation algorithm [15]. Segmentation results demonstrating the performance of the proposed and the Blobworld algorithms are presented in Figures 8 and 9. Although segmentation results are imperfect, as is generally the case with segmentation algorithms, most regions created by the proposed algorithm correspond to a semantic object or a part of one. Even in the latter case, most indexing features (e.g., luminance, color) describing the semantic object appearing in the image can be reliably extracted.

Objective evaluation of segmentation quality was performed using images belonging to various classes and manually generated reference masks (Figures 8 and 9). The employed evaluation criterion is based on the measure of *spatial accuracy* proposed in [47] for foreground/background masks. For the purpose of evaluating still image segmentation results, each reference region g_κ, $\kappa = 1,\dots,K_g$, of the reference mask (ground truth) is associated with a different region s_k of the created segmentation mask on the basis of region overlapping considerations (i.e., s_k is chosen so that $g_\kappa \cap s_k$ is maximized). Then, the spatial accuracy of the segmentation is evaluated by separately considering each reference region as a foreground reference region and applying the criterion of [47] on the pair of $\{g_\kappa, s_k\}$. During this process, all other reference regions are treated as backgrounds. A weighted sum of misclassified pixels for each reference region is the output of this process. The sum of these error measures for all reference regions is used for the objective evaluation of segmentation accuracy; values of the sum closer to zero indicate better segmentation. Numerical evaluation results and comparison using the segmentation masks of Figures 8 and 9 are reported in Table 1.

Following the creation of the region low-level-descriptor database, the mapping between these low-level descriptors and the intermediate-level descriptors defined by the object ontology was performed. This was done by estimating the low-level-descriptor lower and upper boundaries corresponding to each intermediate-level descriptor value, as discussed in Section 3.2. Since a large number of heterogeneous images was used for the initial boundary calculation, future insertion of heterogeneous images to the database is not expected to significantly alter the proportion of image regions associated with each descriptor. Thus, the mapping between low-level and intermediate-level descriptors is not to be repeated, unless the database drastically changes.

The next step in testing with the proposed system was to use the object ontology to define, using the available intermediate-level descriptors/descriptor values, high-level concepts, that is, real-life objects. Since the purpose of the first phase of each query is to employ these definitions to reduce the data set by excluding obviously irrelevant regions, the definitions of semantic objects need not be particularly restrictive (Figure 6). This is convenient from the users' point of view, since the user can not be expected to have perfect knowledge of the color, size, shape, and position characteristics of the sought object.

Subsequently, several experiments were conducted using single-keyword or dual-keyword queries to retrieve images belonging to particular classes, for example, images containing tigers, fireworks, roses, and so forth. In most experiments, class population was 100 images; under-represented classes were also used, with population ranging from 6 to 44 images. Performing ontology-based querying resulted in initial query results being produced by excluding the majority of regions in the database, that were found to be clearly irrelevant. As a result, one or more pages of twenty randomly selected and potentially relevant image regions were presented to the user to be manually evaluated. This resulted in the "relevant" check-box being checked for those that were actually relevant. Usually, evaluating two pages of image regions was found to be sufficient; the average number of image region pages evaluated, when querying for each object class, is presented in Table 2. Note that in all experiments, each query was submitted five times to accommodate for varying performance due to different randomly chosen image sets being presented to the user. The average time required for the SVM training and the subsequent region ranking was 0.12 seconds for single-keyword and 0.3 seconds for dual-keyword

[1] Corel stock photo library, Corel Corporation, Ontario, Canada.

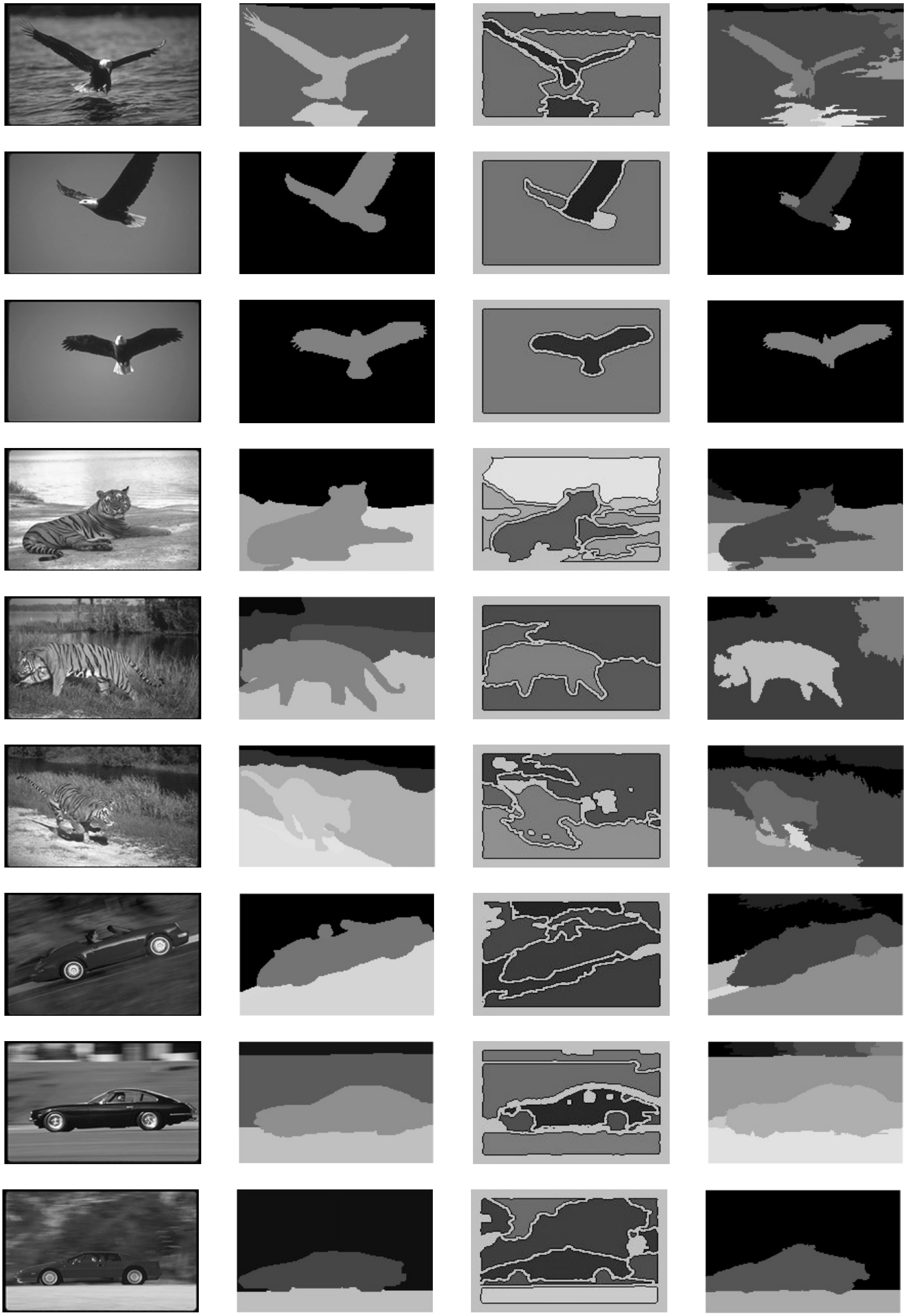


Figure 8: Segmentation results for images belonging to classes eagles, tigers, and cars. Images are shown in the first column, followed by reference masks (second column), results of the Blobworld segmentation algorithm (third column), and results of the proposed algorithm (fourth column).

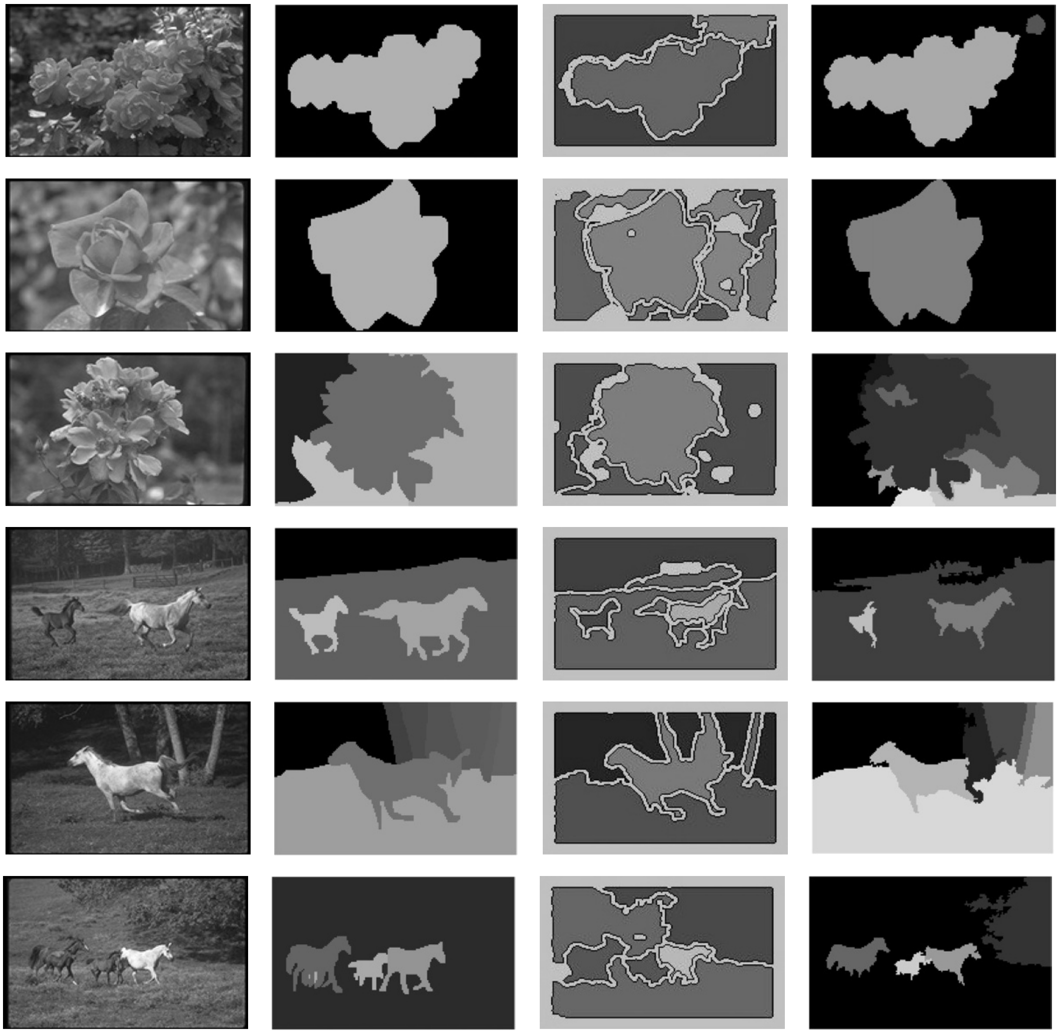


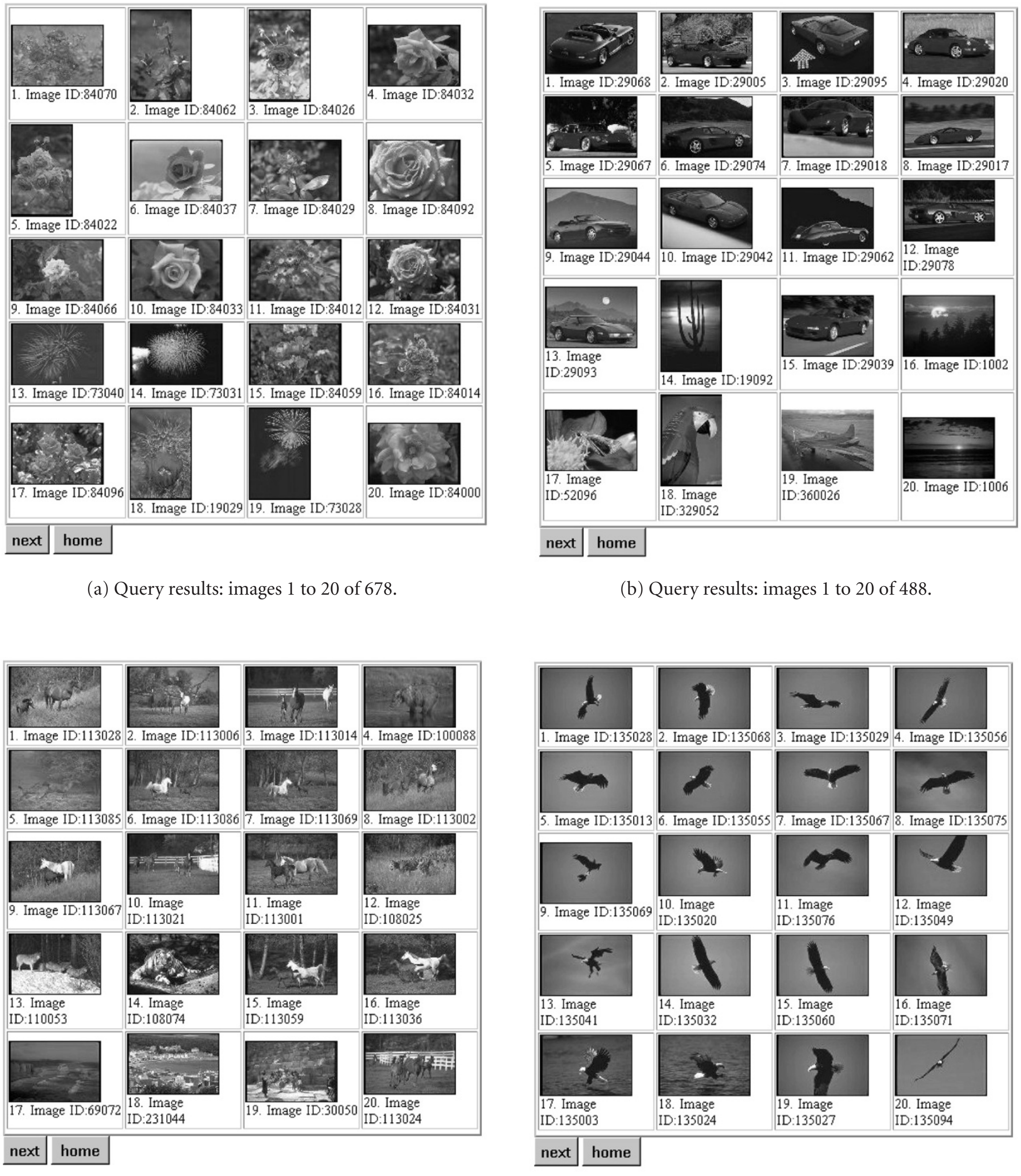
Figure 9: Segmentation results for images belonging to classes roses and horses. Images are shown in the first column, followed by reference masks (second column), results of the Blobworld segmentation algorithm (third column), and results of the proposed algorithm (fourth column).

queries, on a 2 GHz Pentium IV PC. Relevance feedback was then repeated by manually evaluating the regions contained in the first page of the output of the first relevance feedback round. On the average, 0.13 seconds for single-keyword and 0.33 seconds for dual-keyword queries were required for this round. Results after the second round of relevance feedback are presented in Figure 10; precision-recall diagrams for each class of queries after one and two rounds of relevance feedback are presented in Figures 11 and 12. The term *precision* is defined as the fraction of retrieved images which are relevant and the term *recall* as the fraction of relevant images which are retrieved [15].

Table 2: Average number of image region pages evaluated for the first round of relevance feedback.

Query	Pages
Tiger + grass	2.8
Brown horse + grass	2
Bald eagle + blue sky	2.8
Fireworks + black sky	2
Rose	2.2
Sunset	2.2
Red car	2
Yellow car	5.8

In order to further evaluate the above results, experiments were also conducted using the QbE paradigm and global image histograms that were introduced in [48] and used widely ever since. The histograms were based on bins of width 20 in each dimension of the $L^*a^*b^*$ color space. Again, each query was submitted five times, each time using a different and randomly selected key image belonging to the desired class. Comparison (Figures 11 and 12) reveals



(a) Query results: images 1 to 20 of 678.

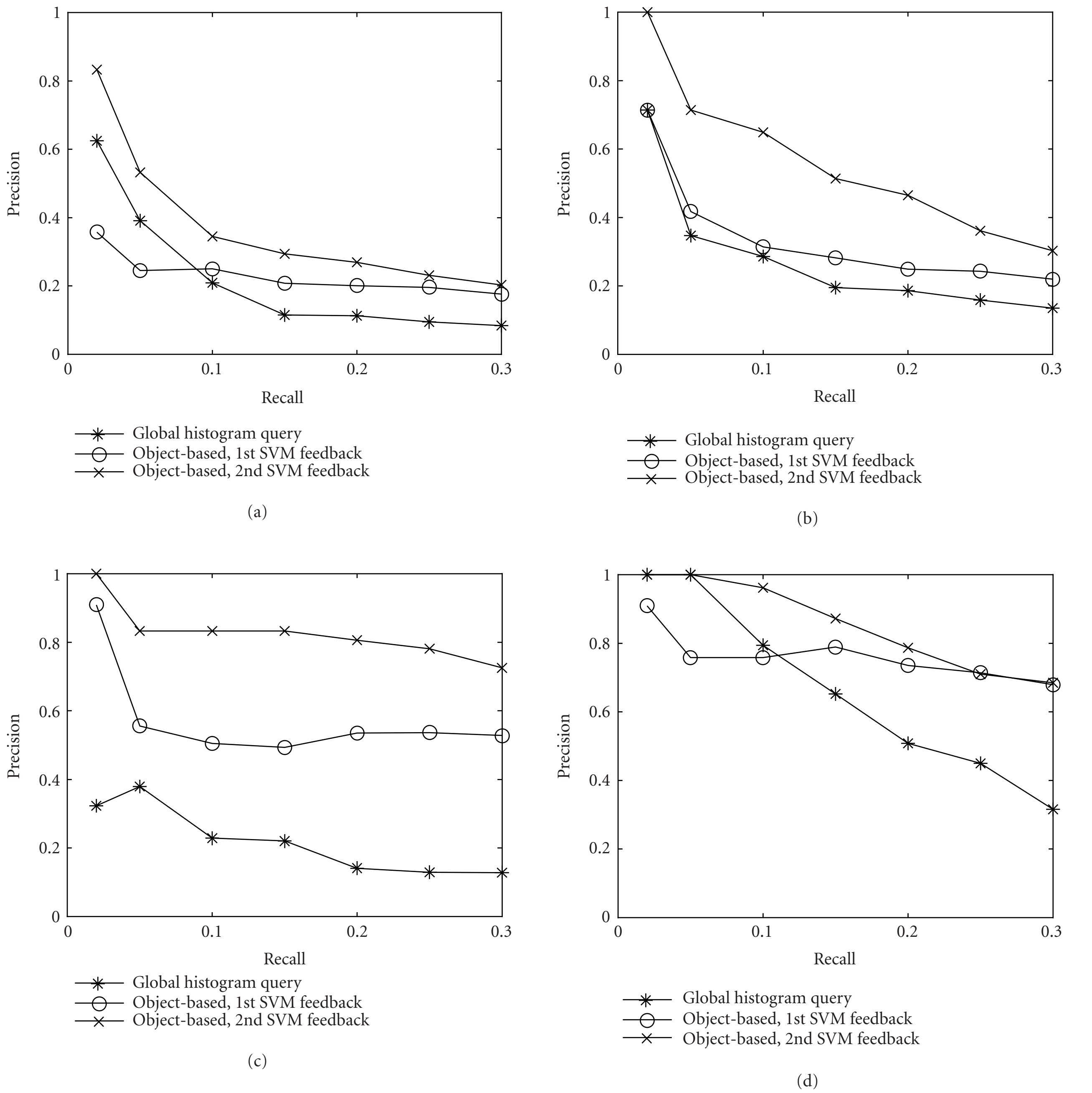
(b) Query results: images 1 to 20 of 488.

(c) Query results: images 1 to 20 of 502.

(d) Query results: images 1 to 20 of 902.

FIGURE 10: Results for single-object queries (a) rose and (b) red car; and dual-object queries (c) brown horse and grass and (d) bald eagle and blue sky, after the second round of relevance feedback.

that even after a single stage of relevance feedback, the proposed method generally yields significantly better results; a second round of relevance feedback leads to further improvement. In Figure 12, results are presented, among others, for a query for a severely under-represented class of objects, *yellow cars*, only 6 images of which are contained in the collection of 5000 images. It can be seen that after the second round of relevance feedback, the proposed scheme performs better than global histograms and manages to rank highly at least one such image, whereas the global histogram method [48] retrieves nothing but the key image used for initiating the query, which already was at the users disposal. Additionally, the diagram calculated for the global histogram method relies on the assumption that it is possible to provide the user



(a)

(b)

(c)

(d)

Figure 11: Precision-recall diagrams for four two-keyword queries and comparison with global histogram method: (a) tiger and grass, (b) brown horse and grass, (c) bald eagle and blue sky, and (d) firework and black sky.

with such a key image, so as to enable the initiation of the query; this is at least debatable when it comes to such under-represented classes.

5. CONCLUSIONS

A methodology was presented in this paper for the flexible and user-friendly retrieval of color images, combining a number of image processing and machine learning tools, such as a time-efficient and unsupervised segmentation algorithm, a simple ontology defining intermediate-level descriptors, and a relevance feedback mechanism based on support vector machines. The resulting methodology is applicable to generic image collections, where no possibility of structuring a domain-specific knowledge base exists. The proposed scheme overcomes the restrictions of conventional methods,

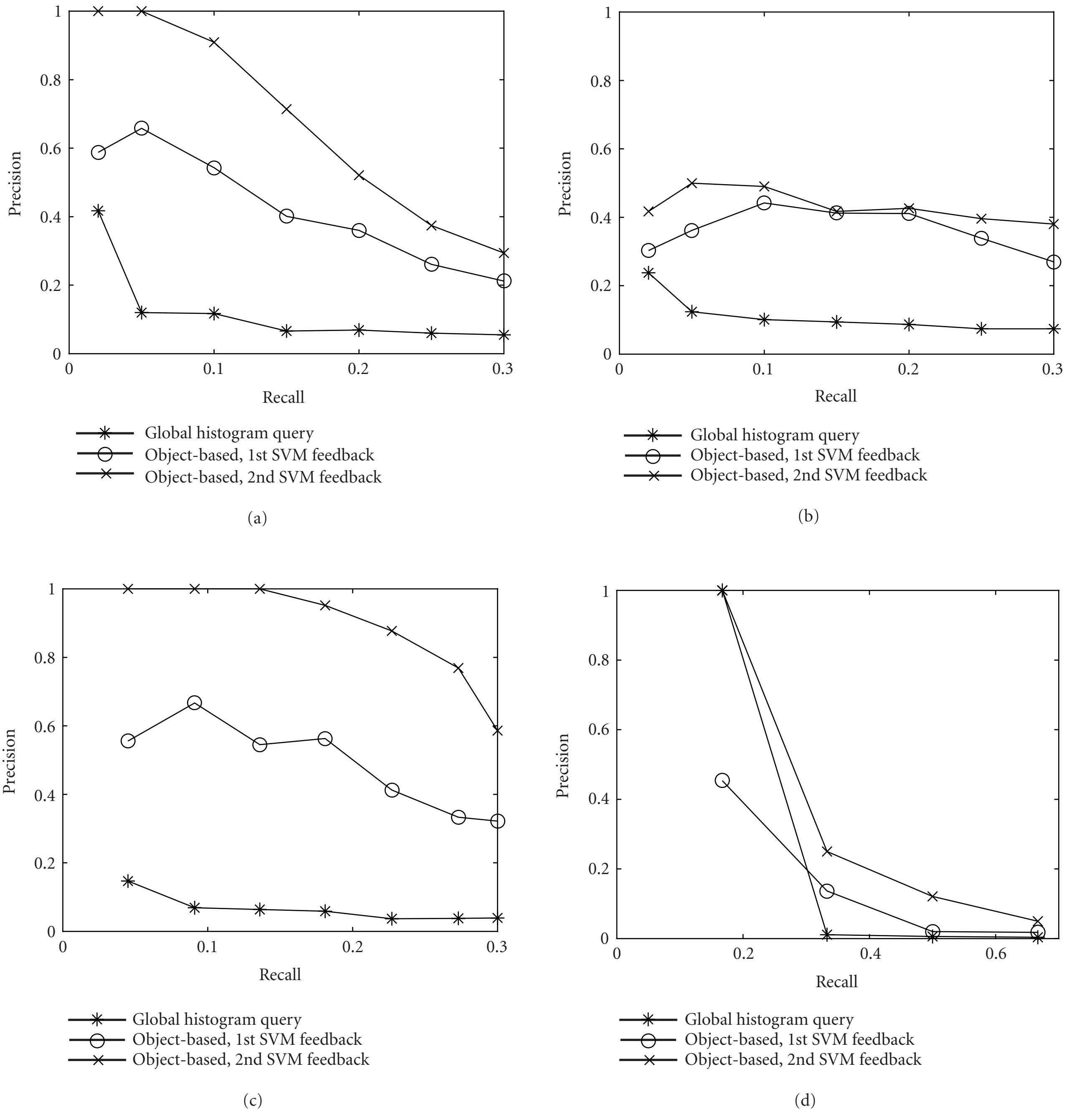


Figure 12: Precision-recall diagrams for four single-keyword queries and comparison with global histogram method. An example of querying for a severely under-represented class of images (yellow car, 6 such images contained in the collection) is also shown. Results are shown for queries (a) rose, (b) sunset, (c) red car, and (d) yellow car.

such as the need for the availability of key images or image captions, and requires no manual tuning of weights, thus offering flexibility and user-friendliness. Experiments conducted on a large collection of images demonstrate the effectiveness of our approach in terms of precision versus recall.

ACKNOWLEDGMENTS

This work was supported by the European Union project "Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval" (IST-2001-32795). The assistance of COST211 quat is also gratefully acknowledged.

REFERENCES

[1] M. R. Naphade and T. S. Huang, "Extracting semantics from audio-visual content: the final frontier in multimedia retrieval," *IEEE Transactions on Neural Networks*, vol. 13, no. 4, pp. 793–810, 2002.

[2] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, 2000.

[3] S. Christodoulakis, M. Theodoridou, F. Ho, M. Papa, and A. Pathria, "Multimedia document presentation, information extraction, and document formation in MINOS: a model and a system," *ACM Trans. Office Information Systems*, vol. 4, no. 4, pp. 345–383, 1986.

[4] C. Zhang and T. Chen, "An active learning framework for content-based information retrieval," *IEEE Trans. Multimedia*, vol. 4, no. 2, pp. 260–268, 2002.

[5] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 602–615, 1998.

[6] C. Faloutsos, R. Barber, M. Flickner, et al., "Efficient and effective querying by image content," *Journal of Intelligent Information Systems*, vol. 3, no. 3/4, pp. 231–262, 1994.

[7] M. Flickner, H. Sawhney, W. Niblack, et al., "Query by image and video content: the QBIC system," *IEEE Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[8] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.

[9] W. Y. Ma and B. S. Manjunath, "NeTra: a toolbox for navigating large image databases," *Multimedia Systems*, vol. 7, no. 3, pp. 184–198, 1999.

[10] T. S. Huang, S. Mehrotra, and K. Ramchandran, "Multimedia analysis and retrieval system (MARS) project," in *Proc. 33rd Annual Clinic on Library Application of Data Processing—Digital Image Access and Retrieval*, Urbana-Champaign, Ill, USA, March 1996.

[11] A. Pentland, R. Picard, and S. Sclaroff, "Photobook: content-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233–254, 1996.

[12] J. R. Smith and S.-F. Chang, "VisualSEEk: a fully automated content-based image query system," in *Proc. ACM Multimedia*, pp. 87–98, ACM Press, Boston, Mass, USA, November 1996.

[13] I. Kompatsiaris, E. Triantafillou, and M. G. Strintzis, "Region-based color image indexing and retrieval," in *Proc. 2001 International Conference on Image Processing*, vol. 1, pp. 658–661, Thessaloniki, Greece, October 2001.

[14] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and texture-based image segmentation using EM and its application to content-based image retrieval," in *Proc. 6th International Conference on Computer Vision*, pp. 675–682, Bombay, India, January 1998.

[15] C. Carson, S. Belongie, H. Greenspan, and J. Malik, "Blobworld: image segmentation using expectation-maximization and its application to image querying," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1026–1038, 2002.

[16] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[17] E. Tuncel and L. Onural, "Utilization of the recursive shortest spanning tree algorithm for video-object segmentation by 2-D affine motion modeling," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 10, no. 5, pp. 776–781, 2000.

[18] H. Gao, W.-C. Siu, and C.-H. Hou, "Improved techniques for automatic image segmentation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1273–1280, 2001.

[19] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[20] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[21] M. R. Naphade, T. Kristjansson, B. Frey, and T. S. Huang, "Probabilistic multimedia objects (multijects): a novel approach to video indexing and retrieval in multimedia systems," in *Proc. IEEE International Conference on Image Processing*, vol. 3, pp. 536–540, Chicago, Ill, USA, October 1998.

[22] Y. Lu, C. Hu, X. Zhu, H. Zhang, and Q. Yang, "A unified framework for semantics and feature based relevance feedback in image retrieval systems," in *Proc. 8th ACM Multimedia*, pp. 31–37, Los Angeles, Calif, USA, October–November 2000.

[23] X. S. Zhou and T. S. Huang, "Unifying keywords and visual contents in image retrieval," *IEEE Multimedia*, vol. 9, no. 2, pp. 23–33, 2002.

[24] A. Yoshitaka and T. Ichikawa, "A survey on content-based retrieval for multimedia databases," *IEEE Trans. Knowledge and Data Engineering*, vol. 11, no. 1, pp. 81–93, 1999.

[25] W. Al-Khatib, Y. F. Day, A. Ghafoor, and P. B. Berra, "Semantic modeling and knowledge representation in multimedia databases," *IEEE Trans. Knowledge and Data Engineering*, vol. 11, no. 1, pp. 64–80, 1999.

[26] G. Tsechpenakis, G. Akrivas, G. Andreou, G. Stamou, and S. Kollias, "Knowledge-assisted video analysis and object detection," in *Proc. European Symposium on Intelligent Technologies, Hybrid Systems and Their Implementation on Smart Adaptive Systems*, Algarve, Portugal, September 2002.

[27] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra, "Relevance feedback: a power tool for interactive content-based image retrieval," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 644–655, 1998.

[28] N. D. Doulamis, A. D. Doulamis, and S. D. Kollias, "Nonlinear relevance feedback: improving the performance of content-based retrieval systems," in *Proc. IEEE International Conference on Multimedia and Expo*, vol. 1, pp. 331–334, New York, NY, USA, August 2000.

[29] V. Kashyap, K. Shah, and A. P. Sheth, "Metadata for building the multimedia patch quilt," in *Multimedia Database Systems: Issues and Research Directions*, pp. 297–319, Springer-Verlag, New York, NY, USA, 1995.

[30] B. Chandrasekaran, J. R. Josephson, and V. R. Benjamins, "What are ontologies, and why do we need them?," *IEEE Intelligent Systems*, vol. 14, no. 1, pp. 20–26, 1999.

[31] S. Staab, R. Studer, H.-P. Schnurr, and Y. Sure, "Knowledge processes and ontologies," *IEEE Intelligent Systems*, vol. 16, no. 1, pp. 26–34, 2001.

[32] P. Martin and P. W. Eklund, "Knowledge retrieval and the World Wide Web," *IEEE Intelligent Systems*, vol. 15, no. 3, pp. 18–25, 2000.

[33] A. Raouzaiou, N. Tsapatsoulis, V. Tzouvaras, G. Stamou, and S. D. Kollias, "A hybrid intelligence system for facial expression recognition," in *Proc. European Symposium on Intelligent Technologies, Hybrid Systems and Their Implementation on Smart Adaptive Systems*, Algarve, Portugal, September 2002.

[34] J. McQueen, "Some methods for classification and analyis of

multivariate observations," in *Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–296, Berkeley, Calif, USA, 1967.

[35] I. Kompatsiaris and M. G. Strintzis, "Spatiotemporal segmentation and tracking of objects for visualization of videoconference image sequences," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 10, no. 8, pp. 1388–1402, 2000.

[36] N. V. Boulgouris, I. Kompatsiaris, V. Mezaris, and M. G. Strintzis, "Content-based watermarking for indexing using robust segmentation," in *Proc. 3rd European Workshop on Image Analysis For Multimedia Interactive Services*, Tampere, Finland, May 2001.

[37] V. Mezaris, I. Kompatsiaris, and M. G. Strintzis, "A framework for the efficient segmentation of large-format color images," in *Proc. IEEE International Conference on Image Processing*, vol. 1, pp. 761–764, Rochester, NY, USA, September 2002.

[38] S. Liapis, E. Sifakis, and G. Tziritas, "Color and/or texture segmentation using deterministic relaxation and fast marching algorithms," in *Proc. IEEE 15th International Conference on Pattern Recognition*, vol. 3, pp. 621–624, Barcelona, Spain, September 2000.

[39] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Processing*, vol. 4, no. 11, pp. 1549–1560, 1995.

[40] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*, Addison-Wesley, Reading, Mass, USA, 1974.

[41] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision*, McGraw Hill, New York, NY, USA, 1995.

[42] A. T. Schreiber, B. Dubbeldam, J. Wielemaker, and B. Wielinga, "Ontology-based photo annotation," *IEEE Intelligent Systems*, vol. 16, no. 3, pp. 66–74, 2001.

[43] E. Bozsak, M. Ehrig, S. Handschuh, et al., "KAON—towards a large scale semantic web," in *Proc. 3rd International Conference on E-Commerce and Web Technologies*, pp. 304–313, Aix-en-Provence, France, September 2002.

[44] G.-D. Guo, A. K. Jain, W.-Y. Ma, and H.-J. Zhang, "Learning similarity measure for natural image retrieval with relevance feedback," *IEEE Transactions on Neural Networks*, vol. 13, no. 4, pp. 811–820, 2002.

[45] V. N. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, NY, USA, 1998.

[46] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," 2001, http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[47] X. Marichal, P. Villegas, and A. Salcedo, "Objective evaluation of segmentation masks in video sequences," in *Proc. Workshop on Image Analysis for Multimedia Interactive Services*, pp. 85–88, Berlin, Germany, May 1999.

[48] M. Swain and D. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

Vasileios Mezaris was born in Athens, Greece, in 1979. He received his Diploma degree in electrical and computer engineering in 2001 from the electrical and computer engineering Department, Aristotle University of Thessaloniki, Greece, where he is currently working towards the Ph.D. degree. He is also a Graduate Research Assistant at the Informatics and Telematics Institute, Thessaloniki, Greece. His research interests include still image segmentation, video segmentation and object tracking, and content-based indexing and retrieval. V. Mezaris is a Member of the IEEE and the Technical Chamber of Greece.

Ioannis Kompatsiaris received his Diploma degree in electrical engineering and his Ph.D. degree in 3D model-based image sequence coding from Aristotle University of Thessaloniki (AUTH), Thessaloniki, Greece in 1996 and 2001, respectively. He is a Senior Researcher at the Informatics and Telematics Institute, Thessaloniki. Prior to his current position, he was a Leading Researcher on 2D and 3D imaging at AUTH. His research interests include 2D and 3D monoscopic and multiview image sequence analysis and coding, semantic annotation of multimedia content, multimedia information retrieval and knowledge discovery, and MPEG-4 and MPEG-7 standards. His involvement with those research areas has led to the coauthoring of 2 book chapters, 13 papers in refereed journals, and more than 40 papers in international conferences. He has served as a regular reviewer for a number of international journals and conferences. Since 1996, he has been involved in more than 13 projects in Greece, funded by the European Commission(EC) and the Greek Ministry of Research and Technology. I. Kompatsiaris is an IEEE Member, a Member of the IEE Visual Information Engineering Technical Advisory Panel, and a Member of the Technical Chamber of Greece.

Michael G. Strintzis received the Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1967, and the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, in 1969 and 1970, respectively. He then joined the Electrical Engineering Department at the University of Pittsburgh, Pittsburgh, Pa, where he served as an Assistant Professor (1970–1976) and an Associate Professor (1976–1980). Since 1980, he has been a Professor of electrical and computer engineering at the University of Thessaloniki, Thessaloniki, Greece, and, since 1999, a Director of the Informatics and Telematics Research Institute, Thessaloniki. His current research interests include 2D and 3D image coding, image processing, biomedical signal and image processing, and DVD and Internet data authentication and copy protection. Dr. Strintzis is serving as Associate Editor for the IEEE Transactions on Circuits and Systems for Video Technology since 1999. In 1984, he was awarded one of the Centennial Medals of the IEEE.