Multimedia Systems 10: 317–331 (2005)
Digital Object Identifier (DOI) 10.1007/s00530-004-0158-z

A structured learning framework for content-based image indexing and visual query

Joo-Hwee Lim[1], **Jesse S. Jin**[2]

[1] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613 (e-mail: joohwee@i2r.a-star.edu.sg)
[2] University of Newcastle, Callaghan NSW 2308, Australia (e-mail: jesse.jin@newcastle.edu.au)

Published online: 12 January 2005 – © Springer-Verlag 2005

Abstract. Nonspecific images in a broad domain remain a challenge for content-based image retrieval. As a typical example, consumer photos exhibit highly varied content, diverse resolutions, and inconsistent quality. The objects are usually ill-posed, occluded, and cluttered with poor lighting, focus, and exposure. Traditional image retrieval approaches face many obstacles such as semantic description of images, robust semantic object segmentation, small sampling problem, semantic gaps between low-level features and high-level semantics, etc.

To manage the high diversity of images in a broad domain, we propose a structured learning framework to systematically design domain-relevant visual semantics, known as semantic support regions, to support index and query in a content-based image retrieval system. Semantic support regions are segmentation-free image regions that exhibit semantic meanings and that can be learned statistically to span a new indexing space. They are detected from image content, reconciled across multiple resolutions, and aggregated spatially to form local semantic histograms. The resulting compact and abstract representation can support both similarity-based query and compositional visual query efficiently. The query by spatial icons (QBSI) formulation is a unique visual query language to explicitly specify visual icons and spatial extents in a Boolean expression.

For empirical evaluation, we perform the learning and indexing processes of 26 semantic support regions over 2400 heterogeneous consumer photos from a single family using Support Vector Machines. We report a 27% improvement in average precision over a very high dimension feature-based approach on 24 semantic queries based on multiple examples and pooled ground truths. Last but not least, we demonstrate the usefulness of the visual query language with 15 QBSI queries that have attained high precision values at top retrieved images on the 2400 consumer images.

Keywords: Content-based image retrieval – Image semantics – Image indexing – Image query

1 Introduction

Retrieving images using their visual content has been a challenge for multimedia research. Early approaches concentrated on low-level features, such as colour, texture, shape, etc. Recent approaches apply image analysis and segmentation to obtain semantic description of images and retrieve images using semantic information. Query methods for the former include *query by example* (QBE) and *query by canvas* (QBC). Query methods for the latter include *query by keywords* (QBK), *query by sketches* (QBS), and *query by spatial icons* (QBSI).

1.1 Query methods

Query by example (QBE) (e.g. QBIC [12], Photobook [32]) requires a relevant image to be visible or available as a query example to start with the search. For example, the ImageRover [45] and Webseek [39] systems deploy text-based queries to obtain an initial set of images, and the PicToSeek [13] approach allows the user to supply a query image. Query by canvas (QBC) (e.g. QBIC [12], Virage [2]) lets the user compose a visual query using geometrical shapes, colours, and textures. This approach inherently tends to specify things/stuff of interest in an indirect way using primitive features. Moreover, the similarity matching between query and images relies on effective presegmentation of regions in the images, which is generally complex and difficult.

Query by keywords (QBK) allows information to be described in high-level meaningful terms. But it cannot be generated automatically by the current content-based image indexing systems. However, manual annotation is usually incomplete, inconsistent, and context sensitive. Moreover, there are situations where image semantics cannot be captured by labelling alone [1]. Query by sketches (QBS) (e.g. [5,10]) lets the user draw the shape of an object as query. But articulating a shape precisely or drawing some ill-defined shapes (e.g. tree, sitting person, mountain) may not be easy. Automatic object shape extraction from cluttered scene images is also an open problem. Hence QBS applications have been limited to images of dominant objects on uniform background [10]. Alternatively, the user is involved in guiding the segmentation during query [9].

Correspondence to: J.-H. Lim

A new query paradigm that allows explicit placement of visual semantics (e.g. face, sky, building, etc.) has been proposed independently [17,22,24]. Unlike the discussed query formulation methods that expect the retrieval system to guess a user's intention expressed implicitly in the query, query by spatial icons (QBSI) lets the user specify a query using higher-level visual semantics represented by visual icons with spatial constraints explicitly in a Boolean expression. In the case of implicit query expression, specifying pool water, sunflowers, or a crowd is unnatural, if not impossible.

1.2 Semantic gap

Low-level features can be easily extracted from images. However, they are not completely descriptive for meaningful retrieval. High-level semantic information is useful and effective in retrieval, but it depends heavily on semantic regions, which are themselves difficult to obtain. Between low-level features and high-level semantic information is an unsolved "semantic gap" [37].

In our opinion, the semantic gap is due to two inherent problems. One problem is that the extraction of complete semantics from image data is extremely hard as it demands general object recognition and scene understanding. This is the *semantic extraction problem*. The other problem is the complexity, ambiguity, and subjectivity in user interpretation, i.e. the *semantic interpretation problem*. The call for user interpretation can occur at three stages, namely prequery (e.g. manual annotation for QBK as discussed above), query, and postquery (e.g. relevance feedback) interventions.

In fact, relevance feedback is regarded as a promising technique for bridging the semantic gap in image retrieval [4,35]. However the correctness of a user's feedback may not be statistically reflected due to the small sampling problem. Though there are innovative techniques proposed for increasing the number of training examples with relevance feedback [46,49], the experimental results are not conclusive yet. An interesting interface model based on guided exploration has also been explored [36]. An inevitable situation that requires user interpretation occurs during query specification. In this paper, we focus on the *semantic interpretation problem* as it relates to query specification.

A unique and promising monotonic tree approach that models scenery images as discrete structural elements has been proposed recently to bridge the semantic gap in content-based image retrieval [41]. Based on simple assumptions about the colour, location, harshness, and shape of scenery features, monotonic trees embody the domain knowledge about scenery images to classify image regions into eight scenery object types with high accuracy to support semantics-based image retrieval.

In this paper, we also aim to bridge the semantic gap, but with different emphases. We study visual semantics that can be directly extracted from image content (without using associated text) with computer vision techniques. In particular, a challenge for computer vision in an unconstrained broad image domain is the usually very large number of object classes in polysemic images. As a typical example of a complex image database [33], consumer photos exhibit highly varied content and imperfect image quality due to the spontaneous and casual nature of image capturing. The objects in consumer images are usually ill-posed, occluded, and cluttered from poor lighting, focus, and exposure. Robust object segmentation for such noisy images is still an unsolved problem [37].

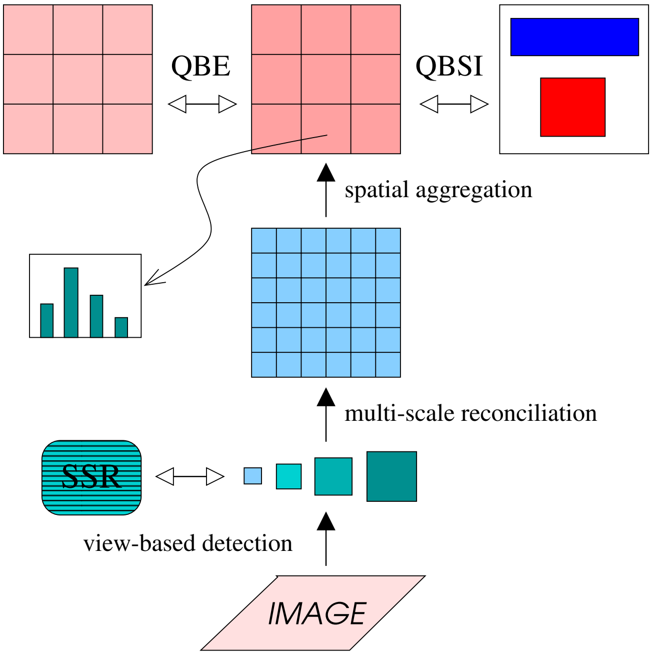

Fig. 1. A structured learning framework for indexing and query

To address the issue of high content diversity, we propose a structured learning framework to facilitate the modular design and extraction of domain-relevant visual semantics, known as *semantic support regions* (SSRs), and hence to deal with larger sets of local visual vocabulary (26 SSRs in our case) in building content-based image retrieval systems. In a nutshell, our proposed framework incorporates modular view-based object detectors to generate spatial semantic signatures for similarity-based and fuzzy-logic-based query processing without region segmentation. Hence our approach is not restricted to images that have a main area of attention, which are assumed by other approaches that attempt object-based indexing and retrieval [27,44].

Semantic support regions are segmentation-free image regions that exhibit semantic meanings and that can be learned statistically to span a new indexing space. They are detected in image content, reconciled across multiple resolutions, and aggregated spatially to form local semantic histograms. The resulting compact and abstract representation can support both similarity-based query and compositional visual query efficiently. Figure 1 summarizes our proposed framework in a schematic diagram. In the figure, arrows with solid heads denote processing steps and arrows with empty heads represent matching.

We apply our method to consumer images that contain highly varied content, diverse resolutions, and inconsistent quality. The significant contribution of the paper is the introduction of the concept of SSRs, which possess the following properties:

- They are extracted directly from images without segmentation and possess semantic power. They can be used to

circumvent the *semantic extraction problem* in the semantic gap.
- Spatial information is retained in the index based on SSRs, so QBSI can be naturally and efficiently applied to alleviate the *semantic interpretation problem* during query.
- SSRs are learned and detected from multiscale tessellated image blocks in a modular manner. The blocks are generally large in number and show statistical significance.

For empirical evaluation, we perform the learning and indexing of 26 SSRs for 2400 heterogeneous consumer photos from a single family using Support Vector Machines. We report a 27% improvement in average precision over a very high dimension feature-based approach on 24 semantic queries based on multiple examples and pooled ground truths. Furthermore, we demonstrate the usefulness of the QBSI with 15 visual queries that have attained high precision values on the top retrieved images from the same test collection of consumer images.

The rest of the paper is organized as follows. In the next section we point out specific research that shares similar objectives. In Sect. 3 we explain how SSRs are learned and used in image indexing. In Sect. 4, we detail query processing for query by multiple examples and spatial icons. Section 5 presents an evaluation of the framework on QBME and QBSI with 2400 genuine consumer photos.

2 Related work

Content-based image retrieval research has progressed from the pioneering feature-based approach (e.g. [2,12,32]) to the region-based approach (e.g. [8,18,38]). However, a desired feature, and hence a key research challenge, is to extract semantics to support meaningful queries. Here we cover recent relevant work in this direction not mentioned in the survey paper ([37, p. 1361]). For comprehensive coverage and understanding of the feature-based and region-based approaches, the reader is referred to the survey paper [37] and to individual papers [2,8,12,18,32,38]. In the case of semantic video indexing, we refer the reader to a recent survey [40] and a representative work [30].

2.1 Semantic extraction and indexing

We first look at related work in semantic extraction and indexing. Town and Sinclair [47] describe a semantic labelling approach to image retrieval. An image is segmented into nonoverlapping regions, and each is classified into 11 visual categories suited to outdoor scenes by artificial neural networks. Both similarity-based matching and region-based matching are supported. The evaluation was carried out on over 1000 Corel images and about 500 home photos, with better classification and retrieval results obtained for the professional Corel images even though the home photo set was smaller than the Corel image set.

A generative approach to segmenting and labelling regions is given in [16]. While generative models offer a modular framework for learning the semantic classes, such models may not work well when the classes have close multimodal distributions, and the data near the discriminative boundary will not be emphasized. In an attempt to classify indoor/outdoor and natural/man-made images, a Bayesian approach was used in [16] to combine class likelihoods resulting from multiresolution probabilistic class labels [7]. The class likelihoods were estimated based on local average colour information and complex wavelet transform coefficients. No further semantic abstraction was performed locally. A total of 480 and 605 images were used as training and test sets, respectively.

Indeed highly accurate segmentation of objects is a major bottleneck for broad domains such as consumer photos except for selected narrow domains when few dominant objects are recorded against a clear background [37]. A key innovation that distinguishes our method from the above systems is that no presegmentation of regions is needed. Instead, 26 SSRs are learned and detected during image indexing from tessellated block-based image regions. Moreover, the local classification decisions are reconciled across multiple resolutions and aggregated over spatial areas as local semantic representations.

The recent monotonic tree approach [41] provides a unique framework for analysing scenery images. Based on a new concept of *monotonic line*, image data are progressively represented as hierarchies of structural elements, which are classified and clustered into semantic regions of sky, building, tree, waves, placid water, lawn, snow, and others with qualifying scores. The qualifying scores for different element categories are computed based on different assumptions about the colour, location, harshness, and shape of scenery features. The scenery features in [41] were tested on 6776 Corel and 1444 PhotoDisc images with very good retrieval results.

From the perspective of using local semantics to bridge the semantic gap, our SSR approach can be viewed in several respects as an extension of the monotonic tree approach. The SSR approach deals with more heterogeneous consumer images using a statistical learning method to automatically map the relationships between second-order statistical local colour and texture features and a larger local visual vocabulary (8 vs. 26). Both the monotonic tree and SSR approaches share similar motivations in multiscale representations (tree structure versus mutliscale detection and reconciliation) and qualifying score computation for a larger image region based on spatial areas. However, this paper proposes and evaluates a new query method that allows explicit specification of semantic elements with spatial constraints.

Motivated as an analogy of the “keywords” of an image, the theories of keyblocks [50] and visual keywords [23,24] also build image indexes from multiresolution image blocks without segmentation. However, the generation of keyblocks or visual keywords is based on either clustering [20,21,23,50] or manual selection [24,50]. While the semantics obtained from unsupervised learning is not strong, the manual selection approach requires intensive human expert labour. Although automatic selection was proposed as an alternative to keyblock generation [50], the codebook-based process is primarily cluster based and may not be discriminative enough for semantic detection.

Interesting and promising efforts have been made recently to associate images with words automatically [3,11,19]. In the case of annotating specific regions [3,11], one major limitation is that the methods rely on semantically meaningful segmentation. For research devoted to automatic annotation of entire images [19], image categories that have visually diverse content (e.g. indoor, street scenes) still present a great challenge for learning.

2.2 Semantic query specification

Next we turn to semantic query specification with spatial constraints. The ImageScape system [17] allows placement of icons (face, sky, water, tree/grass, sand/stone) on a canvas to create a query. However, unlike our approach [22,24], the spatial extent of the icons placed is not emphasized. Moreover, it is not clear in [17] how a query of semantic icons is processed. Last but not least, no proper evaluation has been reported.

As an enhancement to QBE, the query by multiple regions (QBMR) approach [28] allows for a the composition of a query from multiple regions from example images with or without spatial layout. Our QBSI approach can complement the QBMR method in two ways. It is useful when the user is not looking for specific visual similarity but rather more abstract visual concepts. The QBSI interface can also be used to obtain an initial set of relevant images for QBMR as the latter still suffers from the boostrapping problem. Furthermore, the QBSI approach does not need the computation of best matching region and best spatial configuration, as is required by QBMR [28]. The query processing of QBSI, which is based on principled fuzzy operations, is simple and efficient.

Another novel feature in our approach not available in the above works is the hierarchy of visual concepts. That is, SSRs can be structured into an Is-A or a Part-Whole hierarchy for detection, indexing, and query. For example, a sky SSR class can be further divided into subclasses of clear, cloudy, and blue skies with associated specific detectors. A QBSI query can then involve a specific type of sky or a generic sky concept. Another interesting structural mechanism is to detect and index a SSR in terms of its parts (e.g. [29]).

3 Structured learning for image indexing

Semantic support regions (SSRs) are salient image patches that exhibit semantic meanings. A cropped face region, a typical grass patch, a patch of swimming pool water, etc. can all be treated as their instances. In this paper, the SSRs are learned a priori and detected during image indexing from multiscale block-based image regions, as inspired by the multiresolution view-based object recognition framework [31,42], hence without a region segmentation step. The key in image indexing here is not to record the primitive feature vectors themselves but to project them into a classification space spanned by semantic labels and use the soft classification decisions as the local indexes for futher aggregation.

3.1 SSR learning

To compute the SSRs from training instances, we use Support Vector Machines (SVMs) [14]. We extract suitable features such as colour and texture for a local image patch and denote this feature vector by z. A support vector classifier $\mathcal{S}_i$ devoted to a class i of SSR is treated as a function of z, $\mathcal{S}_i(z) \in (-\infty, +\infty)$. Then elements in the classification vector T for region z can be normalized within $[0, 1]$ using the softmax function [6]

$$T_i(z) = \frac{\exp^{\mathcal{S}_i(z)}}{\sum_j \exp^{\mathcal{S}_j(z)}} . \qquad (1)$$

Due to the properties of the softmax function, $T_i(z)$ will never be zero. In this paper, as we regard each SVM as an expert on a SSR class, the outputs of $\mathcal{S}_i \ \ \forall i$ is forced to 0 if there exists some $\mathcal{S}_j, j \neq i$ that has a positive output. More specifically, if there is only one SVM classifier $\mathcal{S}_i$ having positive output, then $T_i(z) = 1$ (and $T_j(z) = 0, j \neq i$). If more than one SVM classifier $\mathcal{S}_i$ has positive outputs, then $T_i(z)$ will be positive values determined by the softmax function, while the other $T_j(z) = 0, j \neq i$. Finally, if all SVM classifers $\mathcal{S}_i \ \ \forall i$ have non-positive outputs, then the values of $T_i(z)$ will be non-zero as computed by the softmax function.

For the experiments described in this paper, since we are dealing with heterogeneous consumer photos, we adopt colour and texture features to characterize SSRs. Hence a feature vector z has two parts, namely a colour feature vector z^c and a texture feature vector z^t. For the colour feature, as the image patch for training and detection is relatively small, the mean and standard deviation of each colour channel are deemed sufficient (i.e. z^c has six dimensions). We use the YIQ colour space over other colour spaces (e.g. RGB, HSV, LUV) as it performed better in our experiments. For the texture feature, we adopted the Gabor coefficients, which have been shown to provide excellent pattern retrieval results [26]. Similarly, the means and standard deviations of the Gabor coefficients (five scales and six orientations) in an image block are computed as z^t, which has 60 dimensions. To normalize both the colour and texture features, we use the Gaussian (i.e. zero-mean) normalization.

The distance or similarity measure depends on the kernel adopted for the SVMs. For the experimental results reported in this paper, we have adopted polynomial kernels. To balance the contributions of the colour and texture features, we have modified the similarity measure $sim(y, z)$ between feature vector y and z as

$$sim(y, z) = \frac{1}{2}\left(\frac{y^c \cdot z^c}{|y^c||z^c|} + \frac{y^t \cdot z^t}{|y^t||z^t|}\right) , \qquad (2)$$

where $y \cdot z$ denotes dot product operation.

For the data set and experiments reported in this paper, we have designed 26 classes of SSRs (i.e. $S_i, i = 1, 2, \cdots, 26$ in Eq. 1). They are organized into 8 superclasses, namely `People`, `Sky`, `Ground`, `Water`, `Foliage`, `Mountain`, `Building`, and `Interior`. Figure 2 shows single examples of these 26 classes of SSRs. This visual vocabulary is decided by 3 human subjects in consensus after studying the test collection.

We cropped 554 image regions from 138 images and used 375 (i.e. two thirds) of them (from 105 images) as training data for SVMs to compute the support vectors of the SSRs and the remaining one third (i.e. 179) as test data for generalization performance. In other words, both the training and test data for SSRs utilize only a small percentage (5.8%) of the 2400-image collection. We experimented with the polynomial and radial basis function kernels with different parameter values. Among all the kernels evaluated, those with better generalization results on the test data were used for the indexing and

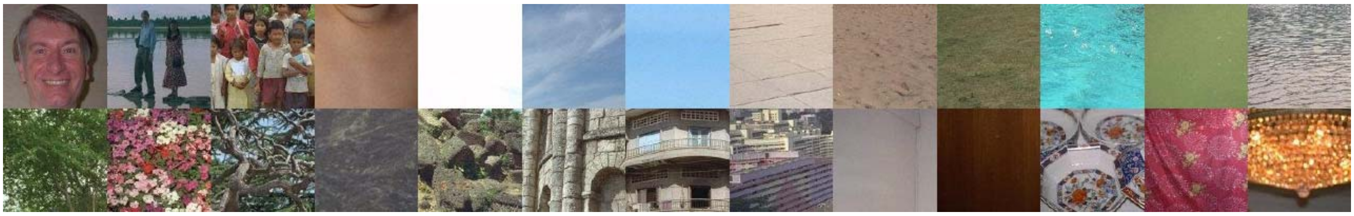

Fig. 2. Examples of semantic support regions (*top down, left* to *right*): `People (Face, Figure, Crowd, Skin), Sky (Clear, Cloudy, Blue), Ground (Floor, Sand, Grass), Water (Pool, Pond, River), Foliage (Green, Floral, Branch), Mountain (Far, Rocky), Building (Old, City, Far), Interior (Wall, Wooden, China, Fabric, Light)`

Table 1. Training statistics of the 26 SSR classes

	Min.	Max.	Avg.
Num. pos. trg.	5	26	14.4
Num. sup. vec.	9	66	33.3
Num. pos. test	3	13	6.9
Num. errors	0	14	5.7
Error (%)	0	7.8	3.2

retrieval tasks. A polynomial kernel with degree 2 and constant 1 ($C = 100$) [14] produced the best results on precision and recall. Hence it was adopted in the rest of our experiments.

Table 1 lists the training statistics of the 26 SSR classes. The columns show, left to right, the minimum, maximum, and average of the number of positive training examples (from a total of 375), the number of support vectors computed from the training examples, the number of positive test examples (from a total of 179), the number of misclassified examples on the 179-region test set, and the percentage of error on the test set. The negative training (test) examples for a SSR class are the union of positive training (test) examples of the other 25 classes. The minimum number of positive training and test examples are from the `Interior:Wooden` SSR, while their maximum numbers are from the `People:Face` class. The mininum and maximum numbers of support vectors are associated with the `Sky:Clear` and `Building:Old` SSRs, respectively. The SSR with the best generalization is the `Interior:Wooden` class, while the worst test error belongs to the `Building:Old` class.

3.2 SSR detection

Once a vocabulary of domain-relevant SSRs has been learned in the form of binary SVMs, an image can be indexed automatically against the SSRs. Figure 3 depicts a three-layer visual information processing architecture for image indexing. The bottom layer denotes the pixel-feature maps computed for feature extraction. In our experiments, conceptually there are 3 colour maps (i.e. YIQ channels) and 30 texture maps (i.e. Gabor coefficients of 5 scales and 6 orientations). From these maps feature vectors z^c and z^t compatible with those adopted for SSR learning (Eq. 2) are extracted.

To detect SSRs with translation and scale invariance in an image to be indexed, the image is scanned with windows of different scales, similar to the strategy in view-based object detection [31]. More precisely, given an image I with resolution $M \times N$, the middle layer (Fig. 3), Reconciled Detection Map (RDM), has a lower resolution of $P \times Q, P \leq M, Q \leq N$. Each pixel (p, q) in RDM corresponds to a two-dimensional region of size $r_x \times r_y$ in I. We further allow tessellation displacements $d_x, d_y > 0$ in X, Y directions, respectively, such that adjacent pixels in RDM along the X direction (along the Y direction) have receptive fields in I that are displaced by d_x pixels along the X direction (d_y pixels along the Y direction) in I. When an image has been scanned, each pixel (p, q) that covers a region z in the pixel-feature layer will consolidate the SSR classification vector $T_i(z)$ (Eq. 1).

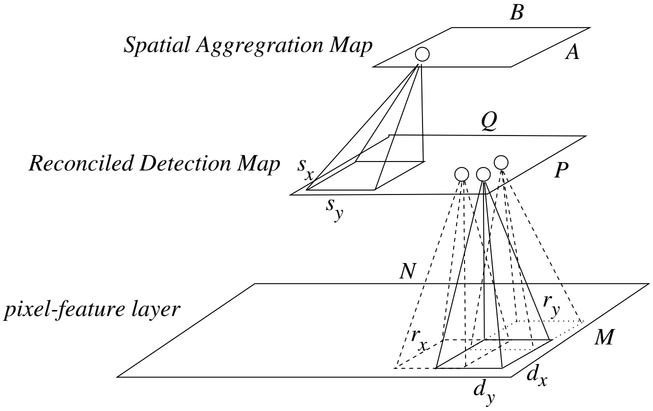

Fig. 3. A visual information processing architecture for image indexing

In our experiments, we progressively increase the window size $r_x \times r_y$ from 20×20 to 60×60 at a displacement (d_x, d_y) of $(10, 10)$ pixels, on a 240×360 size-normalized image. That is, after the detection step, we have five maps of detection of dimensions 23×35 to 19×31, which are reconciled into a common RDM to be explained below.

Using larger images may allow for more accurate features for SVM learning and classification, but the computation requirement is higher. In fact, the strategy adopted in view-based object detection [31, 42] is to fix the window size and resize the image so that it is smaller to achieve multiscale detection. Hence the number of pixels available for object detection is constant. To alleviate the effect of feature extraction on small window size, we fix the image size (after size normalization) and increase the window size instead. As our features z^c and z^t are second-order statistical features (i.e. mean and standard deviation), we do not see any problem with the window sizes

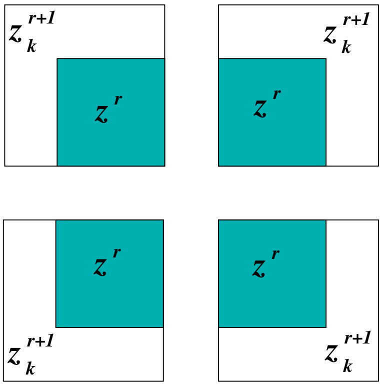

Fig. 4. Reconciling multiscale SSR detection maps

we adopted, as can be seen from the generalization performance shown in Table 1.

3.3 Multiscale reconciliation

In the case of object detection [31,42], the system only needs to output the bounding box of an object detected at any location at any image scale attempted. In our case of image indexing, we seek a common representation of multiple SSRs detected from various image scales attempted. Hence we need to devise a new way to fuse multiscale SSR detection outcomes.

To reconcile the detection maps across different resolutions onto a common basis, we adopt the following principle: If the most confident classification of a region at resolution r is less than that of a larger region (at resolution $r+1$) that subsumes the region, then the classification output of the region should be replaced by those of the larger region at resolution $r+1$. For instance, if the detection of a face is more confident than that of a building at the nose region (assuming "nose" is not in the SSR vocabulary), then the entire region covered by the face, which subsumes the nose region, should be labelled as face.

To illustrate the point, suppose a region at resolution r is covered by four larger regions at resolution $r+1$, as shown in Fig. 4. Let $\rho = max_k max_i T_i(z_k^{r+1})$, where k refers to one of the four larger regions in the case of the example shown in Fig. 4. Then the principle of reconciliation says that if $max_i T_i(z^r) < \rho$, the classification vector $T_i(z^r)\ \ \forall i$ should be replaced by the classification vector $T_i(z_m^{r+1})\ \ \forall i$, where $max_i T_i(z_m^{r+1}) = \rho$.

Using this principle, we compare detection maps of two consecutive resolutions at a time, in descending window sizes (i.e. from windows of 60 × 60 and 50 × 50 to windows of 30 × 30 and 20 × 20). After four cycles of reconciliation, the detection map that is based on the smallest scan window (20 × 20) would have consolidated the detection decisions obtained at other resolutions as the RDM (Fig. 3) for further spatial aggregation.

3.4 Spatial aggregation

The purpose of spatial aggregation is to summarize the reconciled detection outcome in a larger spatial region. Suppose a region Z is comprised of n small equal regions with feature vectors $z_1, z_2, \cdots, z_n$, respectively. To account for the relative proportion of detected SSRs in the spatial area Z, the SSR detection vectors of the RDM are aggregated as

$$T_i(Z) = \frac{1}{n} \sum_k T_i(z_k)\,. \tag{3}$$

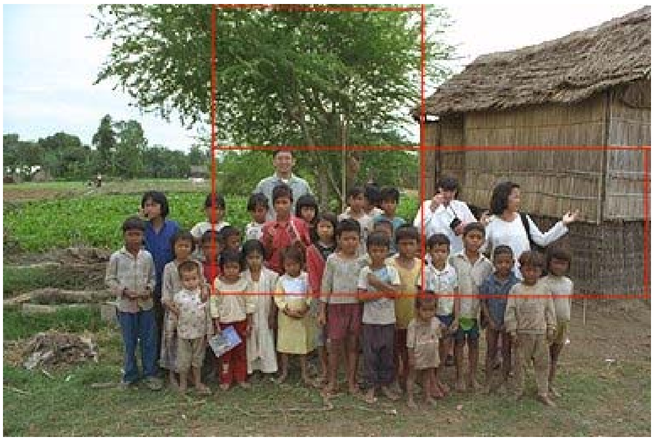

Fig. 5. An example image index

Table 2. Key SSRs recorded as index for image shown in Fig. 5

Image block	Key SSR aggregated	$T_i(Z)$
Top	`Foliage:Green`	0.78
Top	`Foliage:Branch`	0.11
Centre	`People:Crowd`	0.52
Centre	`Foliage:Green`	0.20
Right	`People:Crowd`	0.36
Right	`Building:Old`	0.32

This is illustrated in Fig. 3 where a spatial aggregation map (SAM) further tessellates over RDM with $A \times B, A \leq P, B \leq Q$ pixels. This form of spatial aggregation does not encode the spatial relation explicity. But the design flexibility of s_x, s_y allows us to specify the location and extent in the content to be focused and indexed. We can choose to ignore unimportant areas (e.g. margins) and emphasize certain areas with overlapping tessellation. We can even have different weights attached to the areas during similarity matching (Sect. 4).

The SAM has a representation scheme that is similar to that of local colour histograms, except that the bins refer to proportions of SSRs instead of proportions of colours. They are invariant to translation and rotation about the viewing axis and change only slowly under change of angle of view, change of scale, and occlusion [43]. The effect of averaging in Eq. 3 will not dilute $T_i(Z)$ into a flat histogram. As an illustration, we show the $T_i(Z) \geq 0.1$ of SSRs shown in Fig. 2 in Table 2 for the three tessellated blocks (outlined in red) in Fig. 5. We observe that the dominant $T_i(Z)$ shown capture the content essence in each block with small values distributed in other bins.

3.5 Scalability

Thanks to the modular nature (binary detectors, tessellations, and multiple scales) of our proposed framework, it is straightforward to parallelize the learning, detection, and aggregation tasks. That is, we can train the binary detectors independently. During SSR detection, we compute the feature maps for the pixel-feature layer (Fig. 3) in parallel and feed the combined feature vector to the binary detectors, which can perform classification concurrently. Further parallelization can be achieved by performing SSR detection on different parts of an image (i.e. firing the nodes in RDM simultaneously) and along different scales. After the reconciliation process, which is a sequential process, the spatial aggregation by different nodes in SAM can be carried out concurrently. In short, the indexing process as depicted by Fig. 3 is inherently parallel.

In the current implementation, since we are using two-class SVMs that require both positive and negative examples, retraining of the SVMs is necessary when a new SSR class is added. If we replace two-class SVMs with one-class SVMs [25] or generative models [16], we can train only the new SSR detector based on new positive examples. The performance of one-class SVMs has been shown to be reasonable when compared to other two-class classifiers, though they are rather sensitive to the choice of parameters [25]. The potential problem with generative models was mentioned in Sect. 2.

In general, re-indexing is desirable when the number of SSRs (say s) has been expanded. This is applicable to other indexing methods as well when new feature dimensions are added (e.g. more bins for colour histograms, new feature vector for region segmentation or recognition). However, suppose retraining of existing detectors is not required in the case of one-class SVMs; when a new SSR class $s+1$ has been trained or a better detector becomes available to replace the detector of an existing SSR class j, an efficient re-indexing procedure can be executed as follows. First, SSR detection is performed on all images to be indexed with the new detector ($s+1$ or j) only. The detection outcome [$T_{s+1}(z)$ or $T_j(z)$] is set to either 1 or 0 using a threshold. Next the same reconciliation step can be used to compute the RDM nodes to have either value 1 or 0. Lastly, for each SAM node with a tessellated area Z (size denoted as $|Z|$) in RDM, we count the number (i.e. area) of RDM nodes with value 1 within Z as $|X|$. The new index $T'(Z)$ that includes new SSR detector $s+1$ is computed as

$$T'_{s+1}(Z) = \frac{|X|}{|Z|}, T'_i(Z) = T_i(Z) \cdot \left(1 - \frac{|X|}{|Z|}\right) , \quad (4)$$

and the new index $T'(Z)$ with replacement of the better SSR detector j is revised as

$$T'_j(Z) = \frac{|X|}{|Z|}, T'_{i \neq j}(Z) = \frac{T_i(Z)}{\sum_{i \neq j} T_i(Z)} \cdot \left(1 - \frac{|X|}{|Z|}\right) . \quad (5)$$

4 Query formulation and processing

4.1 Query by multiple examples (QBME)

In QBME, the content-based similarity λ between a query q and an image x can be computed in terms of the similarity between their corresponding local tessellated blocks. For example, the similarity based on L_1 distance measure (city block distance) between query q with m local blocks Y_j and image x with m local blocks Z_j is defined as

$$\lambda(q, x) = 1 - \frac{1}{2m} \sum_j \sum_i |T_i(Y_j) - T_i(Z_j)| . \quad (6)$$

This is equivalent to histogram intersection [43] with further averaging over the number of local histograms m, except that the bins have a semantic interpretation as SSRs. There is a trade-off between content symmetry and spatial specificity. If we want images of similar semantics with different spatial arrangements (e.g. mirror images) to be treated as similar, we can have larger tessellated blocks in SAM (i.e. global histograms). However in applications where spatial locations are considered differentiating, local histograms will provide good sensitivity to spatial specificity. Furthermore, we can attach different weights to the blocks (i.e. Y_j, Z_j) to emphasize the focus of attention (e.g. centre). In this paper, we report experimental results based on even weights as grid tessellation is used. In this paper, we have attempted various similarity and distance measures [e.g. cosine similarity, L_2 distance, Kullback-Leibler (KL) distance, etc.], and the simple city block distance in Eq. 6 has the best performance.

When a query has multiple examples [i.e. query by multiple examples (QBME)], $Q = \{q_1, q_2, \cdots, q_K\}$, the similarity is computed as

$$\lambda(Q, x) = \max_i \lambda(q_i, x) . \quad (7)$$

4.2 Query by spatial icons (QBSI)

A QBSI query is composed as a spatial arrangement of visual semantics. A visual query term (VQT) q specifies a region R where a SSR i should appear, and a query formulus chains these terms up via logical operators. The truth value $\lambda(q, x)$ of a VQT q for any image x is simply defined as

$$\lambda(q, x) = T_i(R) , \quad (8)$$

where $T_i(R)$ is as defined in Eq. 3.

In our current implementation, we support a two-level Is-A hierarchy of SSRs (Fig. 2), though it can be extended to deeper or other forms of hierarchies (e.g. Part-Whole hierarchy). A VQT can involve a specific visual semantics (e.g. swimming pool water, denoted as `Water:Pool`) or a more abstract semantics (e.g. water, denoted as `Water`). On the other hand, the spatial constraint R defines the location and size of the specified visual semantics as drawn on a canvas.

As the visual semantics is learned based on the specific SSR i, the truth value of a VQT that specifies a more abstract visual semantics j (`People, Sky, Ground, Water, Foliage, Mountain, Building`, and `Interior`) is computed as

$$T_j(R) = \max_{i \in V_j} T_i(R) , \quad (9)$$

where V_j denotes the set of classes i that belonged to superclass j.

A QBSI query Q can be specified as a disjunctive normal form of VQT (with or without negation),

$$Q = (q_{11} \wedge q_{12} \wedge \cdots) \vee \cdots \vee (q_{c1} \wedge q_{c2} \wedge \cdots) . \quad (10)$$

Then the query processing of query Q for any image x is to compute the truth value $\lambda(Q, x)$ using appropriate logical operators. As uncertainty values are involved in SSR detection and indexing, we adopt fuzzy operations [15] as follows:

$$\lambda(\bar{q}, x) = 1 - \lambda(q, x) , \quad (11)$$

$$\lambda(q_i \wedge q_j, x) = \min(\lambda(q_i, x), \lambda(q_j, x)) , \quad (12)$$

$$\lambda(q_i \vee q_j, x) = \max(\lambda(q_i, x), \lambda(q_j, x)) . \quad (13)$$

In short, the query processing of QBSI deals with the certainties $T_i(R)$ and $T_j(R)$ of word labels i and j (e.g. `Water:Pool`, `Water`) extracted from image region R. These are abstractions learned upon low-level features such as colour and texture. The indexes no longer store the feature values, and hence the matching does not involve low-level features.

Nevertheless, the vocabulary for QBSI is limited by the semantics that can be learned and detected in image content. For instance, abstract concepts such as “happiness” and “Africa” would require the presence of additional text annotation associated with the images [19], which are not always available in certain application domains (e.g. consumer photos). In this paper, we focus on semantics that can be extracted from the image content alone.

In our existing Web-based prototype, an intuitive graphical interface is provided for a user to specify a QBSI query. To specify a VQT, the user first selects a SSR (specific or abstract) from a palette of icons associated with the SSR. Then a spatial image region based on the selected icon can be drawn by clicking and dragging a rectangular box in a canvas. If the user wishes to apply a negation operator, he or she can click on the NOT button followed by the drawn region. A yellow cross will be superimposed on the selected region. The user can continue to specify more VQTs in a conjunct by repeating the above steps. The user can also start a new conjunct in the disjunctive normal form (Eq. 10) by clicking on the OR button to bring up a new window with canvas and icons. A reset button is provided to clear all the icons drawn for a conjunct in a given window. A typical screen shot is given in Fig. 6 (note that only a subset of the visual icons is displayed in this prototype).

As the region specified by a VQT is arbitrary, the precise computation of $T_i(R)$ using Eq. 3 on reconciled small regions z_k is not cost effective in terms of speed and storage. Hence as a trade-off in our implementation, we pre-indexed the images using a uniform 3×3 spatial tessellation with the 26 SSRs defined in Fig. 2 based on Eqs. 1 and 3. The truth value of a VQT q with region R and SSR i is approximated as

$$\lambda(q, x) = \frac{\sum_{Z_j \in Z} T_i(Z_j)}{|Z|} , \quad (14)$$

where Z consists of any of the 3×3 blocks that has more than half of its area covered by region R.

Another QBSI interface that corresponds to the 3×3 indexing grid is also supported. That is, the user can click on an icon associated with a SSR and fill any block in the grid with the selected icon. In a similar way, a negation operator (NOT button) can be applied to a block (which will be crossed in yellow) and a new window with grid and icons can be invoked (OR button) to start a new conjunct.

5 Experimental results

5.1 Test collection

In this paper, we have decided to evaluate our proposed structured learning framework on unconstrained consumer images. Unlike professional images, which are well defined, carefully taken, and clearly layered, or domain-specific images such as medical images, which have a clear classification and are usually attached with semantic annotation, consumer image content varies significantly, as we highlighted at the beginning of the paper. More often than not, there is no dominant homogeneous colour or texture regions, which poses great difficulty for image segmentation. Moreover, very few consumers will annotate their photos. Hence we cannot assume availability of text for content association. We believe that consumer images are more challenging than the Corel images used by many image retrieval researchers. For example, the region-based matching approach described in [47] was evaluated on over 1000 Corel Photo Library images and about 500 home photos, with better classification and retrieval results obtained for the professional Corel images, even though the home photo set is smaller than the Corel image set.

In this paper, we evaluate our proposed approach on 2400 heterogeneous consumer photos from a single family. These genuine consumer photos were taken over 5 years in several countries in both indoor and outdoor settings. The images are those of the smallest resolution (i.e. 256 × 384) from Kodak PhotoCDs, in both portrait and landscape layouts. After the removal of possibly noisy marginal pixels, the images are of size 240×360. The indexing process automatically detects the layout and applies the corresponding tessellation template. On one hand, the small size of the images allows for more efficient processing. On the other hand, they pose a greater challenge for feature extraction and SSR detection.

To have a feel of the content diversity in our 2400-image collection, we show 72 (3%) of them in Fig. 7. For outdoor images, the content varies from natural landscape (beach, lakeside, river, pond, park, forest, garden, mountain, rocky area, etc.) to city scenes (urban area, rural area, crowded street, market, road with vehicles, swimming pool, temple, mosque, castle, etc.) from different countries and cultures (Singapore, France, China, Cambodia, Malaysia, Indonesia, etc.). The indoor images are taken with different focuses (portrait of single person or a few people, groups of different sizes, people eating, cultural performance, wedding ceremony, interior layout, display of objects like painting, toys, antique collection, etc.). In both outdoor and indoor images, the subject of focus could be people (or faces in photo frame), statues, animals, flowers, buildings (or their miniature in theme park), etc. and their mixture with occlusion, taken with different postures, during the day or at night, from different viewpoints, and at different distances. Figure 8 illustrates some of the photos of bad quality (e.g. faded, overexposed, blurred, dark, etc.). We did not remove these bad-quality photos from our test collection

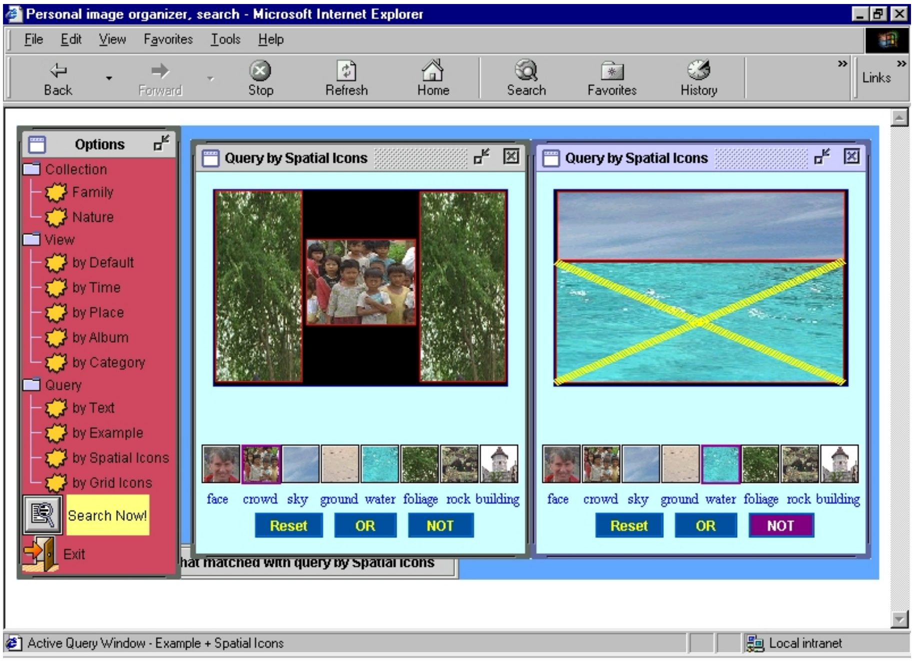
Personal image organizer, search - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Forward
Stop
Refresh
Home
Search
Favorites
History
Links
Options
Collection
Family
Nature
View
by Default
by Time
by Place
by Album
by Category
Query
by Text
by Example
by Spatial Icons
by Grid Icons
Search Now!
Exit
Query by Spatial Icons
face crowd sky ground water foliage rock building
Reset
OR
NOT
Query by Spatial Icons
face crowd sky ground water foliage rock building
Reset
OR
NOT
hat matched with query by Spatial Icons
Active Query Window - Example + Spatial Icons
Local intranet

Fig. 6. A screen shot for QBSI interface: a disjunct of two conjuncts, one with 3 visual query terms (*left*) and the other with 2 visual query terms (*right*), one of which is a negation on water

because we wanted to reflect the complexity of the original data.

As part of our project, the QBSI interface shown in Fig. 6 is one of the query functions provided in our operational prototype, which is implemented in C and Java with Microsoft Access. Our Web-based system also allows query by examples, query by text, query by mixture of query modes, browsing along different dimensions (time, place, people, categories), data management (e.g. addition, deletion, copying of photos and albums), text annotation, SMIL-based [51] slideshow authoring and presentation with music. Last but not least, separate tools are also provided for uploading images to the Web, visual queries (QBME and QBSI), and slideshow presentations on PocketPC.

5.2 QBME experiments

Based on the consensus of 3 human subjects, 24 semantic queries and their ground truths (GT) among the 2400 photos are constructed (Table 3). That is, for each query, every human subject has to look through the entire collection to build the list of relevant images. Note that Q17 and Q18 are not restricted to indoor images, while Q14–Q16 are. Q22 includes both indoor (in building) and outdoor (near building) images. In fact, Fig. 7 shows, in top-down left-to-right manner, three relevant images for queries Q01–24. As we can see from these sample images, the relevant images for any query considered here exhibit highly varied and complex visual appearances. Hence to represent each query, the three human subjects selected three (i.e. $K = 3$ in Eq. 7) relevant photos as query examples for our experiments because a single query image is far from satisfactory for capturing the semantic of any query. Indeed single query images resulted in poor precisions and recalls in our initial experiments. The precisions and recalls were computed without the query images themselves in the lists of retrieved images.

Table 3. The 24 semantic queries used in our QBME experiments

Query	Description	GT
Q01	Indoor	994
Q02	Outdoor	1218
Q03	People eating	76
Q04	People indoors	840
Q05	Interior or object	134
Q06	City scene	697
Q07	Nature scene	521
Q08	At a swimming pool	52
Q09	Street or roadside	645
Q10	Near water	150
Q11	In a park or garden	304
Q12	Near mountain	67
Q13	Buildings	239
Q14	People close up, indoors	73
Q15	Small group, indoors	491
Q16	Large group, indoors	45
Q17	People (mid-range)	277
Q18	People (close-up)	104
Q19	People near water	61
Q20	People, with foliage	259
Q21	People near mountain	35
Q22	People near or in a building	1517
Q23	Garden with flowers	19
Q24	Mountain (far view)	35

In our experiments, we compared our SSR approach (denoted “SSR”) with the feature-based approach that combines colour and texture in a linearly optimal way (denoted “CTO”). We do not compare our approach with other approaches such

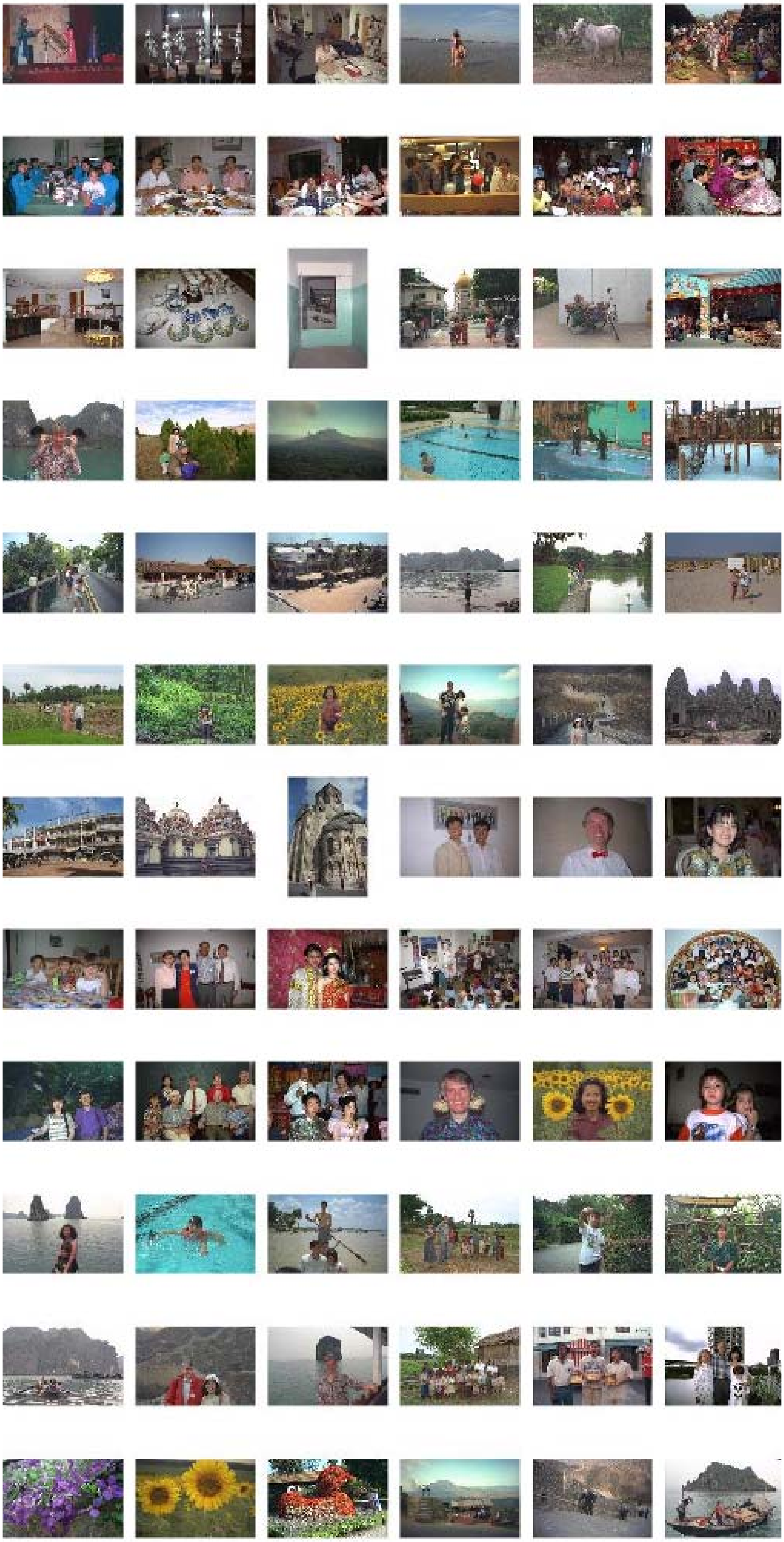

Fig. 7. Sample consumer photos associated with queries 01 to 24

Fig. 8. Some consumer photos of bad quality

as region-based matching here as our initial attempt with region segmentation using 500 outdoor images [48] does not scale up on the 2400-image collection. Indeed, very high dimensions are required for the CTO approach to produce reasonable performance as we shall see below. We adopted similar colour and texture features for the CTO approach to demonstrate the advantage gained from the abstraction layer of SSRs. For the CTO approach, we conducted experiments with various system parameters and selected their best performances. We looked at both the overall average precisions (denoted P_{avg}) and average precisions on the top 30 retrieved images (denoted P_{30}) over 24 queries to select the best performance.

For the colour-based signature, both global and local (4×4 grid) colour histograms of b^3 ($b = 4, 5, \cdots, 17$) number of bins in the RGB colour space were computed on an image. In the case of global colour histograms, the performance saturated at 4096 ($b = 16$) and 4913 ($b = 17$) bins with $P_{avg} = 0.31$ and $P_{30} = 0.48$. Hence the one that used fewer bins was preferred. Among the local colour histograms attempted, the one with 2197 bins ($b = 13$) gave the best precisions with $P_{avg} = 0.32$ and $P_{30} = 0.49$. These performance figures show that more bins are required for a larger image region when the colour distribution is potentially richer in our heterogeneous consumer photo collection. Histogram intersection [43] was used to compare two colour histograms.

For the texture-based signature, we adopted the means and standard deviations of Gabor coeffients and the associated distance measure as reported in [26]. The Gabor coefficients were computed with 5 scales and 6 orientations. Convolution windows of $20 \times 20, 30 \times 30, \cdots, 60 \times 60$ were attempted. Similarly, both global and local (4×4 grid) signatures were experimented with. The best results were obtained when 20×20 windows were used. We obtained $P_{avg} = 0.20$ and $P_{30} = 0.25$ for global signatures and $P_{avg} = 0.21$ and $P_{30} = 0.32$ for local signatures. These inferior results when compared to those of colour histograms led us to conclude that simple statistical texture descriptor is less effective than colour histograms for heterogeneous image content.

Based on the performances of global and local signatures of colour and texture, we decided to fuse their local signatures, which have better precisions. The distance measures between a query and an image for the colour and texture methods were normalized within $[0, 1]$ and combined linearly. Among the relative weights attempted at 0.1 intervals, the best fusion was obtained with $P_{avg} = 0.33$ and $P_{30} = 0.50$, with a 0.8 of colour influence and 0.2 of texture influence. The same performance values (up to two decimal points) were obtained when a multiplicative fusion operator was used.

In our experiments, the tessellation for detection of SSRs was a 4×4 grid of rectangular regions. We compared our SSR approach with the best CTO result (fusion of local colour and texture signatures). Table 4 shows the average precisions (over 24 queries) among the top 20, 30, 50, and 100 retrieved images as well as over all recall points for the methods compared. The results show that our SSR approach outperforms CTO as a whole by nine precision points (a significant 27% improvement) and produces more relevant images on top retrieved images.

Table 4. Average precisions on top retrieved images for QBME experiments

Avg. prec.	CTO	SSR	%
At 20	0.56	0.69	23
At 30	0.50	0.63	26
At 50	0.46	0.56	22
At 100	0.39	0.47	21
Overall	*0.33*	*0.42*	*27*

The experiments were conducted on a Pentium IV PC (1.4 GHz, 256 MB memory). The learning of 26 SSRs on 375 training samples was very fast (less than a minute). The indexing of one image with the SSR approach required about 20 s (without any code optimization), four times that of the CTO approach (local colour and texture histograms). However, the small footprint of SSR signatures is highly efficient in storage space and retrieval. Suppose a 4-byte floating point number is required for each $T_i(Z)$. Then a SSR image index requires less than 2 KB ($26 \times 16 \times 4$) of storage and simple operations on a small number of vectors. Compared to the high-dimension features required by the CTO index ($2000^+ \times 16 \times 4$), there is almost a $100\times$ reduction. In fact, the actual storage size can be further reduced as most of the $T_i(Z)$ entries are zeroes. This would have great advantage over the need to represent and process very high dimensions of colour and texture features and yet not achieve the same level of retrieval performance.

In summary, the image signatures based on SSRs realize semantic abstraction via prior learning and detection of visual classes when compared to direct indexing based on low-level features. The compact representation also resulted in better performance than the optimal fusion of very high dimensions of colour and texture features in our QBME experiments using 24 semantic queries on 2400 heterogeneous consumer photos. Hence we feel that the computational resources devoted to prior learning of SSR and their detection during indexing are a good trade-off for concise semantic representation as well as effective and efficient retrieval performance.

5.3 QBSI experiments

To further evaluate the effectiveness of the local semantic regions indexed for the 2400 consumer photos, we have designed 15 QBSI queries as illustrated in Figs. 9–13.

While queries 01 to 04 focus on single VQTs, queries 05 to 15 demonstrate multiple VQTs. In particular, query 06 is composed to look for indoor images with close-ups of people. Query 07 specifies faces in 3 different regions to enforce ‘small group of people’. Query 10 intends to retrieve images

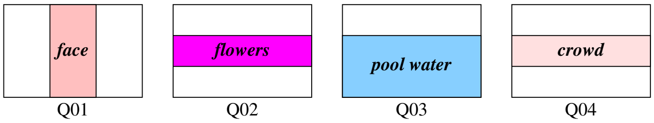

Fig. 9. QBSI queries 01 to 04

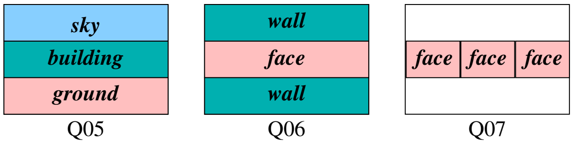

Fig. 10. QBSI queries 05 to 07

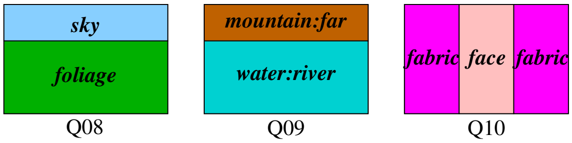

Fig. 11. QBSI queries 08 to 10

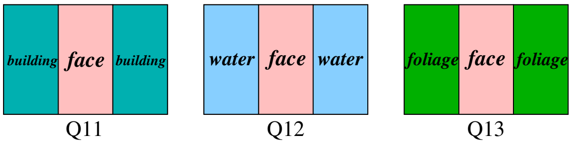

Fig. 12. QBSI queries 11 and 13

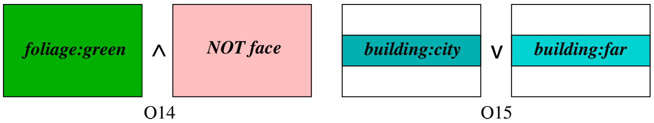

Fig. 13. QBSI queries 14 and 15

related to wedding events whereby auspicious fabric can be seen. Query 14 shows the use of the negation operator. Last but not least, query 15 illustrates the usefulness of the disjunct operator. All the queries except 05 and 08 involve specific SSRs. Queries 05 and 08 are based on superclasses of SSRs. Queries 11 to 13 illustrate the flexibility of mixing SSR (face) and the superclasses (building, water, and foliage). Our SSR indexing framework supports queries with different levels of visual semantics and their mixture.

The indexes are computed based on Eqs. 1 and 3 with face detection enhancement [34]. With our modular framework, the replacement of object detection decisions is simple as described in Sect. 3.5.

Table 5 lists the number of relevant images among the top 20 and 30 retrieved images as well as the size of the ground

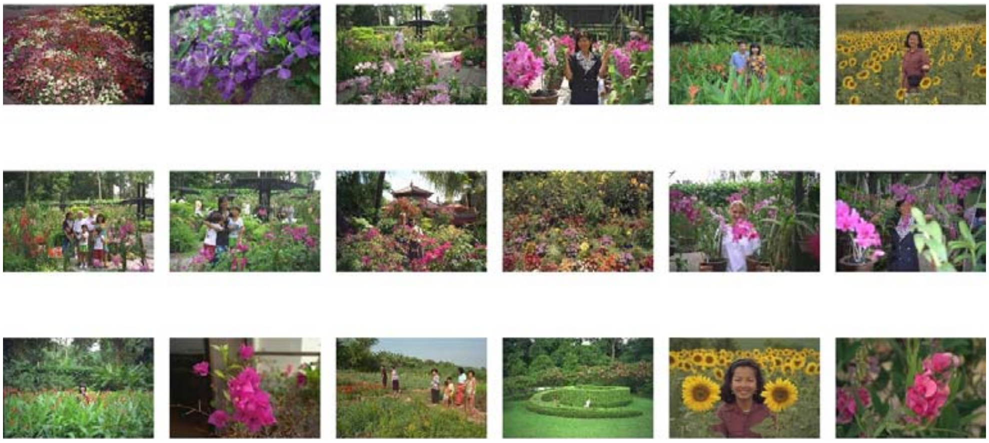

Fig. 14. Top 18 retrieved images for QBSI query 02

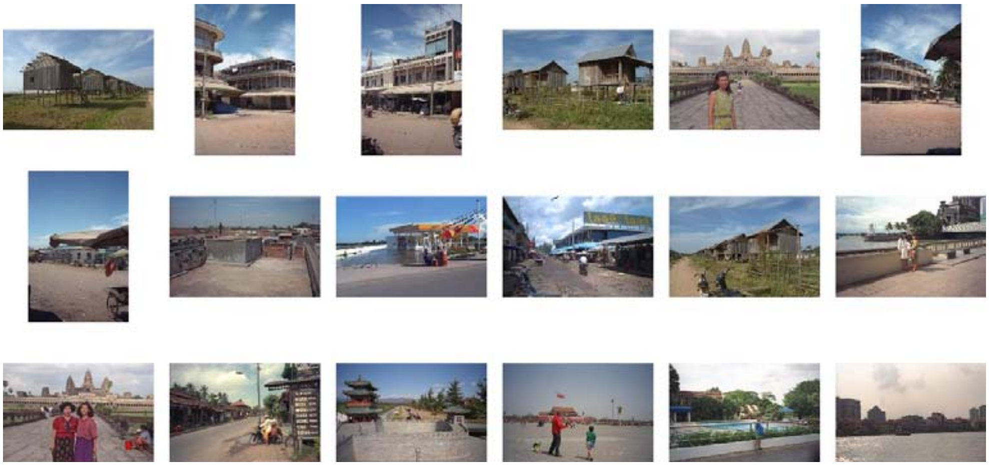

Fig. 15. Top 18 retrieved images for QBSI query 05

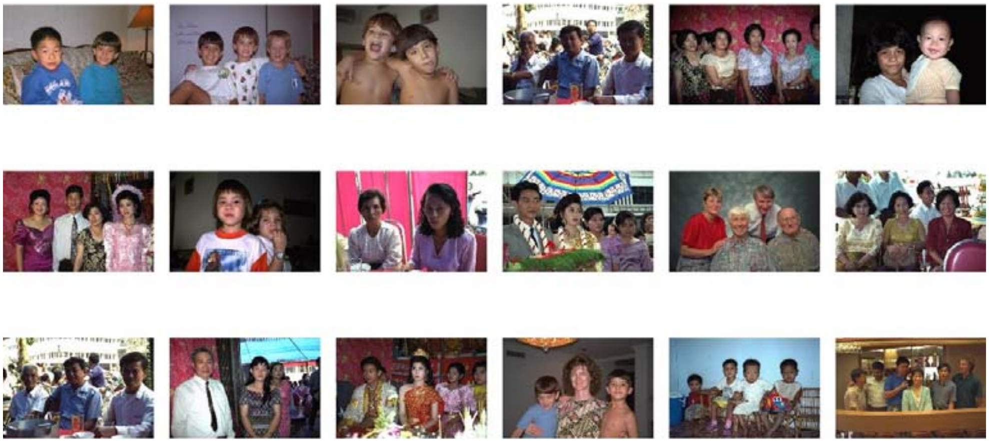

Fig. 16. Top 18 retrieved images for QBSI query 07

Table 5. Precisions on top retrieved images for QBSI experiment

Query	Top 20	Top 30	GT
Q01	14	24	590
Q02	18	23	26
Q03	14	16	44
Q04	16	19	78
Q05	19	26	281
Q06	14	20	302
Q07	20	20	380
Q08	18	25	83
Q09	12	16	19
Q10	14	17	112
Q11	16	25	523
Q12	11	16	61
Q13	18	25	259
Q14	18	25	107
Q15	15	20	234
Avg.	*15.8*	*21.1*	

truth (GT) for each of the queries tested. As shown in the table, the average precisions for the top 20 and 30 retrieved images are 0.79 and 0.70, respectively, which we consider effective for practical applications. Interestingly, queries 02 and 09 demand small numbers of specific images (i.e. around 1%) to be found among 2400 images. The recall among the top 30 retrieved images is high with recall values of 0.88 and 0.84, respectively.

Next we show the top retrieved images for 3 of the 15 queries, namely queries 02, 05, and 07, in Figs. 14, 15, and 16 respectively. In the figures, the top 18 images retrieved are shown in top-down, left-to-right order of decreasing relevance.

For query 02 (Fig. 9), the intention was to look for images with flowers (cf. `Foliage:Floral` in Fig. 2) at the centre. Of the top 18 images shown in Fig. 14, only image 15 is irrelevant as the flower regions is considered too small.

With query 05 (Fig. 10), we look for images with a spatial layout of sky, building, and ground (cf. Fig. 2). Only the last image in Fig. 15 is a false positive where the greyish water was incorrectly detected as ground.

In the case of query 07 (Fig. 10) that looks for small groups of people appearing at the centre of an image (cf. `People:Face` in Fig. 2), the top 18 images shown in Fig. 16 are all found in the GT list for the query.

Compared to existing query formulation methods, our QBSI approach allows explicit specification of visual semantics as illustrated by the 15 queries in Figs. 9–13. Consider the case of query by canvas (QBC). How would a user express visual concepts such as flowers, faces, and buildings using colour and texture or their combination? Query by sketches (QBS) is not very useful either as the shapes of flowers, faces, sky, water, etc. are ill-defined. Compared to the ImageScape system [17] that also allows placement of visual icons as query, our QBSI approach has richer expressive power as we support spatial constraints (Q01 to Q15), negation (Q14), disjunction (Q15), and concept hierarchy (Q05, Q08, Q11–13).

6 Conclusion and future work

In this paper, we have presented an adaptive view-based detection approach to indexing and querying images based on semantic regions. More specifically, our contributions can be listed as follows.

- The proposed SSR approach provides a systematic framework to index image content based on local semantics learned from domain examples. The modular framework also allows new and better view-based object detectors to be embedded easily to enhance retrieval performance, as illustrated by the face detector in our experiments.
- A novel indexing algorithm detects, reconciles, and aggregates SSRs in an image to form semantic histograms without the need for robust region segmentation.
- A comprehensive empirical evaluation has been carried out with 24 semantic queries on 2400 complex images to verify the usefulness of the proposed framework against a typical feature-fusion approach.
- A unique visual query language and processing has been shown to support intuitive and semantic query formulation, which are not available in existing systems, using 15 QBSI queries on 2400 consumer photos.

In the coming months, we would like to apply the framework to other content domains such as medical images and integrate with other semantic sources such as text [3,19]. We are also investigating means to reduce the extent of supervision in learning while retaining a high degree of semantic interpretation.

Acknowledgements. The authors wish to thank J.L. Lebrun for lending his family photos, CMU for their face detector, and T. Joachims for his great SVM^{light} software.

References

1. Armitage L, Enser P (1997) Analysis of user need in image archives. J Inf Sci 23(4):287–299
2. Bach J R, Fuller C, Gupta A, Hampapur A, Horowitz B, Humphrey R, Jain R C, Shu C (1996) Virage image search engine: an open framework for image management. In: Storage and Retrieval for Image and Video Databases IV, Proc. SPIE 2670, pp 76–87
3. Barnard K, Duygulu P, Freitas ND, Forsyth D, Blei D, Jordan MI (2003) Matching words and pictures. J Mach Learn Res 3:1107–1135
4. Cox I J, Miller M L, Minka T P, Papathomas T, Yianilos PN (2000) The Bayesian image retrieval system, PicHunter: theory, implementation and psychophysical experiments. IEEE Trans Image Process 9:20–37
5. Del Bimbo A, Pala P (1997) Visual image retrieval by elastic matching of user sketches. IEEE Trans Pattern Anal Mach Intell 19:121–132
6. Bishop CM (1995) Neural networks for pattern recognition. Clarendon Press, Oxford
7. Bradshaw B (2000) Semantic based image retrieval: a probabilistic approach. In: Proc. ACM Multimedia'2000, pp 167–176
8. Carson C, Belongie S, Greenspan H, Malik J (2002) Blobworld: image segmentation using expectation-maximization and its application to image querying. IEEE Trans Pattern Anal Mach Intell 24(8):1026–1038
9. Cinque L, Lecca F, Levialdi S, Tanimoto S L (2000) Retrieval of images using rich-region descriptions. J Vis Lang Comput 11:303–321
10. Daoudi M, Matusiak S (2000) Visual image retrieval by multiscale description of user sketches. J Vis Lang Comput 11:287–301
11. Duygulu P, Barnard K, de Freitas N, Forsyth D (2002) Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In: Proc. ECCV'2002, pp 97–112
12. Flickner M, Sawhney H, Niblack W, Ashley J, Huang Q, Dom B, Gorkani M, Hafner J, Lee D, Petkovic D, Steele D, Yanker P (1995) Query by image and video content: the QBIC system. IEEE Comput 28(9):23–32
13. Gevers T, Smeulders A (1997) PicToSeek: a content-based image search system for the World Wide Web. In: Proc. Visual 97, pp 93–100
14. Joachims T (1999) Making large-scale SVM learning practical. In: Scholkopf B, Burges C, Smola A (eds) Advances in kernel methods – support vector learning. MIT Press, Cambridge, MA
15. Klir GJ, Folger T (1992) Fuzzy sets, uncertainty, A, information. Prentice Hall, Upper Saddle River, NJ
16. Kumar S, Loui AC, Hebert M (2002) Probabilistic classification of image regions using an observation-constrained generative approach. In: 1st international workshop on generative-model-based vision
17. Lew M (2000) Next-generation web searches for visual content. IEEE Comput 33(11):46–52
18. Li J, Wang JZ Wiederhold G (2000) Integrated region matching for image retrieval. In: Proc. ACM Multimedia'2000, pp 147–156
19. Li J, Wang JZ (2003) Automatic linguistic indexing of pictures by a statistical modeling approach. IEEE Trans Pattern Anal Mach Intell 25(10):1–14
20. Lim JH (1999) Learnable visual keywords for image classification. In: Proc. ACM Digital Libraries, pp 139–145
21. Lim JH (1999) Learning visual keywords for content-based retrieval. In: Proc. IEEE ICMCS, pp 169–173
22. Lim JH (2000) Explicit query formulation with visual keywords. In: Proc. ACM Multimedia'2000, pp 407–409
23. Lim JH (2000) Visual keywords: from text IR to multimedia IR. In: Crestani F, Pasi G (eds) Soft computing in information retrieval: techniques and applications, Physica, Springer, Berlin Heidelberg New York, pp 77–101
24. Lim JH (2001) Building visual vocabulary for image indexation and query formulation. Pattern Anal Appl 4(2/3):125–139
25. Manevitz LM, Yousef M (2001) One-class SVMs for document classification. J Mach Learn Res 2:139–154
26. Manjunath BS, Ma WY (1996) Texture features for browsing and retrieval of image data. IEEE Trans Pattern Anal Mach Intell 18(8):837–842
27. Martinez AM, Serra JR (2000) A new approach to object-related image retrieval. J Vis Lang Comput 11:345–363
28. Moghaddam B, Biermann H, Margaritis D (2001) Regions-of-interest and spatial layout for content-based image retrieval. Multimedia Tools Appl 14:201–210
29. Mohan A, Papageorgiou C, Poggio T (2001) Example-based object detection in images by components. IEEE Trans Pattern Anal Mach Intell 23(4):349–361
30. Naphade MR, Kozintsev IV, Huang TS (2002) A factor graph framework for semantic video indexing. IEEE Trans CSVT 12(1):40–52
31. Papageorgiou PC, Oren M, Poggio T (1997) A general framework for object detection. In: Proc. international conference on computer vision, pp 555–562
32. Pentland A, Picard RW, Sclaroff S (1995) Photobook: content-based manipulation of image databases. Int J Comput Vis 18(3):233–254
33. Rao A, Srihari R, Zhu L, Zhang A (2002) A theory for measuring the complexity of image databases. IEEE Trans Multimedia 4(2):160–173
34. Rowley HA, Baluja S, Kanade T (1998) Neural network-based face detection. IEEE Trans Pattern Anal Mach Intell 20(1):23–38
35. Rui Y, Huang TS, Mehrotra S (1997) Content-based image retrieval with relevance feedback in MARS. In: Proc. IEEE international conference on image processing, pp 815–818
36. Santini S, Gupta A, Jain R (2001) Emergent semantics through interaction in image databases. IEEE Trans Knowl Data Eng 13(3):337–351
37. Smeulders AWM, Worring M, Santini S, Gupta A, Jain R (2000) Content-based image retrieval at the end of the early years. IEEE Trans Pattern Anal Mach Intell 22(12):1349–1380
38. Smith JR, Chang S-F (1996) VisualSEEk: a fully automated content-based image query system. In: Proc. ACM Multimedia 96, Boston
39. Smith JR, Chang S-F (1997) Visually searching the web for content. IEEE Multimedia 4(3):12–20
40. Snoek CGM, Worring M (2005) Multimodal video indexing: a review of the state-of-the-art. Multimedia Tools Appl 25(1):5–35
41. Song Y, Zhang A (2003) Analyzing scenery images by monotonic tree. Multimedia Syst 8(6):495–511

42. Sung KK, Poggio T (1998) Example-based learning for view-based human face detection. IEEE Trans Pattern Anal Mach Intell 20(1):39–51
43. Swain MJ, Ballard DN (1991) Color indexing. Int J Comput Vis 7(1):11–32
44. Tao Y, Grosky WI (2000) Image indexing and retrieval using object-based point feature maps. J Vis Lang Comput 11:323–343
45. Taycher L, Cascia M, Sclaroff S (1997) Image digestion and relevance feedback in the ImageRover WWW search engine. In: Proc. Visual 97, pp 85–91
46. Tieu K, Viola P (2000) Boosting image retrieval. In: Proc. CVPR'2000, pp 1228–1235
47. Town C, Sinclair D (2000) Content-based image retrieval using semantic visual categories. Technical Report 2000.14, AT&T Research, Cambridge, MA
48. Wu JK, Lim JH, Hong DZ (2000) Toward semantics level indexing and retrieval of images and video. In: Proc. 2000 RWC symposium, Tokyo, 17–19 January 2000, pp 159–164
49. Wu Y, Tian Q, Huang TS (2000) Discriminant-EM algorithm with application to image retrieval. In: Proc. CVPR'2000, pp 1222–1227
50. Zhu L, Rao AB, Zhang AD (2002) Theory of keyblock-based image retrieval. ACM Trans Inf Syst 20:224–257
51. W3C: Synchronized Multimedia Integration Language (SMIL 2.0). http://www.w3.org/TR/smil20/