**Image databases:** Conventional database management systems have been designed for managing numeric and text data, which are single dimensional in nature. But spatial or pictorial data are necessarily two or three-dimensional and contain a considerable amount of information that cannot be stored directly as a set of linear relationships. The spatial or two dimensional image data have a great deal of implicit and explicit knowledge. Explicitly, there is sensory information pertaining to the lighting-brightness, shadows, colors, etc. Implicit in all the spatial data is the information related to the concept of position as well as the notion of distance. What makes the management of spatial data very complicated is that the same image can be viewed in different perspectives by various users. Consequently, the primary features that need to be recognized and extracted can be different for each problem domain. Moreover there is a need to be able to represent image data of various kinds. For example, a geographical map would directly pertain to two-dimensional information, whereas a machine design drawing contains information about three-dimensional objects. In addition, in order to support a large variety of continuous and discrete representations, an image database provides access to special-purpose functions for image processing such as primitives for feature extraction. In order to make progress in image databases from the software engineering perspective, it is important to examine innovations in both processing and representations of images. One of the most important problems to be considered in the design of image database systems is how images are stored in an image database. Many data structures have been proposed. Some are pixel-oriented; some utilize quadtrees, or R-trees, and some are vector based. To make an image database system more intelligent, more flexible and an efficient data structure should be used. Also, the knowledge embedded in images should be captured by the data structure as much as possible, especially spatial knowledge. Extracting information from images is a time consuming process. On the other hand, if information or knowledge is extracted from images item by item in advance and stored for later retrieval, we need much more storage capacity and therefore, retrieval would take a long time. In pictorial information retrieval, many approaches have been proposed and include relational database queries, query-by-view [412], quadtrees etc. We now describe some image database systems for which such approaches have been proposed and prototyped.

1. Intelligent Image Database System (IIDS) [103]: This is a prototype intelligent image database system that is based on a new pictorial data structure. Specifically, a new way of representing a picture by a 2-D string has been introduced in IIDS. A picture query can also be specified as a 2-D string. The problem of pictorial information retrieval then becomes one of 2-D string subsequence matching. This approach aJlows an efficient and natural way to construct iconic indexes for pictures. The 2-D string representation is ideally suited to formulating picture queries. The iconic index can not only be used in pictorial information retrieval, but also provides an efficient means for picture browsing. The corresponding 2-D string for this kind of query contains two special icons. One is called single variable icon, which can match any single object; the other is called multivariable icon, which can match any set of objects. In order to increase the power of the IIDS, it would be advantageous to add attributes to symbols in the 2-D strings. In this way, we could convey not only the relative positions of objects, but also information such as orientation, size, and other characteristics.
2. Image Database System (IDB) [591]: Most information management systems are designed to handle traditional alphanumeric data. Today technology makes available resources that allow the management of new classes of information, such as image and voice. When dealing with images, we must generalize the input, processing, and output phases that characterize the management of traditional data types. Furthermore, all these activities require suitable hardware and software instruments. From the end-user point of view, the interface between users and systems would be much more attractive if it were possible to use images to manage images as well as we use words to manage traditional information. Two kinds of data must be managed by an image system: image files and their descriptions. The former are characterized by large sizes and unstructured forms, while the latter have small sizes and structured forms. Images and descriptions are stored on different kinds of devices. The main requirement for image data is the availability of a large memory at low cost. Optical disks can meet this requirement, and image data are now increasingly stored on those special devices, while descriptive information continues to be stored on magnetic disks. The IDB system exploits images as a vehicle of interaction with the user; index images play a fundamental role in completing the selection of images from the archive. The architecture is characterized by modularity and flexibility; each single module is related to a specific task to be performed during the image management process. Functions have been integrated by distributing resources among the nodes of a LAN; each node Corresponds to a workstation and many users can work with the system. Future extensions of IDB will involve the integration of new kinds of information such as audio data and image animation. A hypermedia approach is also being evaluated.
3. Map Database: A map database management system contains facilities to create, modify, store, and retrieve spatial information. A Map Database System (MDS) goes beyond simply replacing paper maps. MDS allows users to view, compare, and analyze spatial relationships. Map databases allow the generation of maps that contain only the information required by the map user. The map information is divided into different layers which overlay on the same area. Typical layers include streams, cities, sewers, roads, highways, secondary streets, water pipes, gas lines, telephone cables and so on. The information in layers may also contain per capital income, product consumption, or other thematic information. Map databases contain large amounts of data. Efficient encoding of the graphical information into a format suitable for digital storage is required. The non-graphical attributes are usually stored using the normal methods. Several different encoding methods are polygon encoding, dual independent map encoding, and 2-D encoding [236].

**Access methods into IDBs:**

**Content based image retrieval (CBIR):**

The pivotal point in content-based retrieval is that the user seeks semantic similarity, but the database can only provide similarity by data processing. This is what we called the semantic gap. At the same time, the sensory gap between the properties in an image and the properties of the object plays a limiting role in retrieving the content of the image.

***Image domain and sensory gap***: In the repertoire of images under consideration - the image domain ***I*** there is a gradual distinction between narrow and broad domains. At one end of the spectrum, we have the narrow domain:

*A narrow domain has a limited and predictable variability in all relevant aspects of its appearance. In a narrow domain, one finds a limited variability of the content of the images.*

Usually, the recording circumstances are also similar over the whole domain. In the narrow domain of lithographs, for instance, the recording is under white light with frontal view and no occlusion. The domain would be wider had the faces been photographed from a crowd or from an outdoor scene. In that case, variations in illumination, clutter in the scene, occlusion, and viewpoint will have a major impact on the analysis. On the other end of the spectrum, we have the broad domain:

*A broad domain has an unlimited and unpredictable variability in its appearance even for the same semantic meaning.*

In broad domains, images are polysemic and their semantics are described only partially. It might be the case that there are conspicuous objects in the scene for which the object class is unknown or even that the interpretation of the scene is not unique. Many problems of practical interest have an image domain in between these extreme ends of the spectrum, see Fig. 4. The notions of broad and narrow domains are helpful in characterizing patterns of use, in selecting features, and in designing systems. In a broad image domain, the gap between the feature description and the semantic interpretation is generally wide. For narrow, specialized image domains, the gap between features and their semantic interpretation are usually smaller, so domain-specific models may help. For faces, many geometric models have been suggested, as well as statistical models. These computational models are not available for broad image domains as the required number of computational variables would be enormous. For broad image domains in particular, one has to resort to generally valid principles. Is the illumination of the domain white or colored? Does it assume defined and fully visible objects or may the scene contain clutter and occluded objects? Is it a 2D-recording of a 2D-scene or a 2D-recording of a 3D-scene? The given characteristics of illumination, presence or absence of occlusion, clutter, and differences in camera viewpoint determine demands on the retrieval methods. The sensory gap is the gap between the object in the world and the information in a (computational) description derived from a recording of that scene. The sensory gap makes the description of objects an ill posed problem: It yields uncertainty in what is known about the state of the object. The sensory gap is particularly poignant when a precise knowledge of the recording conditions is missing. The 2D-records of different 3D-objects can be identical. Without further knowledge, one has to decide that they might represent the same object. Also, a 2D-recording of a 3D-scene contains information accidental for that scene and that sensing but one does not know what part of the information is scene related. The uncertainty due to the sensory gap not only holds for the viewpoint, but also for occlusion (where essential parts telling two objects apart may be out of sight), clutter, and illumination. Comparing alternative interpretations can attenuate the sensory gap. Content-based image retrieval systems may provide support in this disambiguation through elimination among several potential explanations, much the same as in natural language processing.



It is important to establish that content-based retrieval does not rely on describing the content of the image in its entirety. It may be sufficient that a retrieval system presents similar images, similar in some user-defined sense. The description of content should serve that goal primarily. We consider the description of content in two steps. First, we discuss image-processing operations that transpose the image data into another spatial data array, see Fig. 6. We divide the methods over local color, the local texture, or local geometry. They may be characterized in general by: 

where i(x) is the image, element of image space I, g is an operator on images, and the resulting image field is given by f(x). Computational parameters of g may include the size of the neighborhood around x to compute f(x) or a homogeneity criterion when the size of the patch to compute f(x) depends on the actual data. So, the purpose of image processing in image retrieval must be to enhance aspects in the image data relevant to the query and to reduce the remaining aspects. One such goal can be met by using invariance as a tool to deal with accidental distortions in the information introduced by the sensory gap. From the above discussion on the sensory gap, it is clear that invariant features may carry more object-specific information than other features as they are insensitive to the accidental conditions of the sensing.



The degree of invariance, that is, the dimensionality of the group W, should be tailored to the recording circumstances. In general, a feature with a very wide class of invariance loses the power to discriminate among essential differences. The size of the class of images considered equivalent grows with the dimensionality of W. In the end, the invariance may be so wide that no discrimination among objects is retained. The aim is to select the tightest set of invariants suited for the expected set of non-constant conditions.

***Color image processing:*** Color makes the image i(x) take values in a color vector space. The interest in color may be ascribed to the superior discriminating potentiality of a three-dimensional domain compared to the single dimensional domain of gray-level images.

Two aspects of color return in many of the contributions. One is that the recorded color varies considerably with the orientation of the surface, the viewpoint of the camera, the position of the illumination, the spectrum of the illuminant, and the way the light interacts with the object. This variability should be dealt with in one way or another. Second, the human perception of color is an intricate topic where many attempts have been made to capture perceptual similarity. Only when there is no variation in the recording or in the perception is the RGB color representation a good choice since that representation was designed to match the input channel of the eye. RGB-representations are in wide-spread use. They describe the image in its literal color properties. An image expressed as R(x); G(x); B(x) makes most sense when recording in the absence of variance, as is the case, e.g., for art paintings, the color composition of photographs, and trademarks, where two-dimensional images are recorded in frontal view under standard conditions. A significant improvement over the RGB-color space (at least for retrieval applications) comes from the use of opponent color representations, which uses the opponent color axes (R - G; 2B - R - G; R + G + B). This representation has the advantage of isolating the brightness information on the third axis. The HSV-representation is often selected for its invariant properties. The hue is invariant under the orientation of the object with respect to the illumination and camera direction and hence more suited for object retrieval. A wide variety of tight photometric color invariants for object retrieval were derived from an analysis of the Schafer model of object reflection. They derive for matte patches under white light the invariant color space



only dependent on sensor and surface albedo. For a shiny surface and white illumination, they derive the invariant representation as



and two more permutations. The color models are robust against major viewpoint distortion.

***Image processing for local shape:*** Under the name “local shape”, we collect all properties that capture conspicuous geometric details in the image. We prefer the name local shape over differential geometrical properties to express the result rather than the method. The result of local shape evaluation is a dense image data field different from object shape. Local shape characteristics derived from directional color. Scale space theory was devised as the complete and unique primary step in pre-attentive vision, capturing all conspicuous information. It provides the theoretical basis for the detection of conspicuous details on any scale.

***Image texture processing:*** texture is defined as all what is left after color and local shape have been considered or it is defined by such terms as structure and randomness. Many common textures are composed of small textons usually too great in number to be perceived as isolated objects. The elements can be placed more or less regularly or randomly. They can be almost identical or subject to large variations in their appearance and pose. In the context of image retrieval, research is mostly directed toward statistical or generative methods for the characterization of patches. Basic texture properties include the Markovian analysis. In retrieval, the property is computed in a sliding mask for localization. Another important texture analysis technique uses multi-scale autoregressive MRSAR-models, which consider texture as the outcome of a deterministic dynamic system subject to state and observation noise. Other models exploit statistical regularities in the texture field. Wavelets have received wide attention. They have often been considered for their locality and their compression efficiency. Many wavelet transforms are generated by groups of dilations or dilations and rotations that have been said to have some semantic correspondent. The lowest levels of the wavelet transforms have been applied to texture representation sometimes in conjunction with Markovian analysis. Texture search proved useful in satellite images and images of documents. Textures also served as a support feature for segmentation-based recognition, but the texture properties discussed so far offer little semantic referent. They are therefore ill-suited for retrieval applications in which the user wants to use verbal descriptions of the image.

***Description of content: Features (Grouping data)***



In content-based image retrieval, the image is often divided in parts before features are computed from each part, see Fig. 7. Partitionings of the image aim at obtaining more selective features by selecting pixels in a trade-off against having more information in features when no subdivision of the image is used at all. We distinguish the following partitionings:

* When searching for an object, it would be most advantageous to do a complete object segmentation first: “Strong segmentation is a division of the image data into regions in such a way that region T contains the pixels of the silhouette of object O in the real world and nothing else, specified by: T = O.” It should be noted immediately that object segmentation for broad domains of general images is not likely to succeed, with a possible exception for sophisticated techniques in very narrow domains.
* The difficulty of achieving strong segmentation may be circumvented by weak segmentation where grouping is based on data-driven properties: “Weak segmentation is a grouping of the image data in conspicuous regions T internally homogenous according to some criterion, hopefully with T $⊂ $O.”

The criterion is satisfied if region T is within the bounds of object O, but there is no guarantee that the region covers all of the object's area. When the image contains two nearly identical objects close to each other, the weak segmentation algorithm may falsely observe just one patch. Fortunately, in content-based retrieval, this type of error is rarely obstructive for the goal. When occlusion is present in the image, weak segmentation is the best one can hope for. Weak segmentation is used in many retrieval systems, either as a purpose of its own or as a preprocessing stage for data-driven model-based object segmentation.

* When the object has a (nearly) fixed shape, like a traffic light or an eye, we call it a sign:
“Localizing signs is finding an object with a fixed shape and semantic meaning, with T = xcenter.”

Signs are helpful in content-based retrieval as they deliver an immediate and unique semantic interpretation.

* The weakest form of grouping is partitioning: “A partitioning is a division of the data array regardless of the data, symbolized by: T $\ne $O.”

The area T may be the entire image or a conventional partitioning as the central part of the image against the upper, right, left, and lower parts. The feasibility of fixed partitioning comes from the fact that images are created in accordance with certain canons or normative rules, such as placing the horizon about 2/3 up in the picture or keeping the main subject in the central area. This rule is often violated, but this violation in itself has semantic significance. Another possibility of partitioning is to divide the image in tiles of equal size and summarize the dominant feature values in each tile. Each of these four approaches to partitioning leads to a preferred type of features, as summarized in Fig. 8 and illustrated in Fig. 9, where feature hierarchies are used to make a combination on all types.



***Global and Accumulating Features***

In the computational process, features are calculated next. The general class of accumulating features aggregates the spatial information of a partitioning irrespective of the image data. A special type of accumulative features is the global features which are calculated from the entire image. Accumulating features are symbolized by:







represents a fixed tiling of the image. The operator h may hold relative weights, for example, to compute transform coefficients. A simple but very effective approach to accumulating features is to use the histogram, that is, the set of features F(m) ordered by histogram index m. The original idea to use histograms for retrieval comes from Swain and Ballard, who realized that the power to identify an object using color is much larger than that of a gray-valued image. As a histogram loses all information about the location of an object in the image, project the histogram back into the image to locate it by searching for best matches. A histogram may be effective for retrieval as long as there is uniqueness in the color pattern held against the pattern in the rest of the entire data set. In addition, the histogram shows an obvious robustness to translation of the object and rotation about the viewing axis. Swain and Ballard also argue that color histograms change slowly with change in viewpoint and scale and with occlusion.

Another alternative is to add a dimension representing the local distance. This is the correlogram, defined as a three dimensional histogram where the colors of any pair are along the first and second dimension and the spatial distance between them along the third. The auto correlogram defining the distances between pixels of identical colors is found on the diagonal of the correlogram. A more general version is the geometric histogram, with the normal histogram, the correlogram, and several alternatives as special cases. This also includes the histogram of the triangular pixel values, reported to outperform all of the above as it contains more information.

A different view on accumulative features is to demand that all information (or all relevant information) in the image is preserved in the feature values. When the bit content of the features is less than the original image, this boils down to compression transforms. Many compression transforms are known, but the quest is for transforms simultaneously suited as retrieval features. As proper querying for similarity is based on a suitable distance function between images, the transform has to be applied on a metric space. The components of the transform have to correspond to semantically meaningful characteristics of the image. Finally, the transform should admit indexing in compressed form yielding a big computational advantage over having the image be untransformed first

***Salient Features:*** Another way to avoid the brittleness of strong segmentation is to opt for weak segmentation. This leads to a grouping of the data into homogeneous regions. From the merged regions, a selection is made on their salience. The most conspicuous regions are stored. The limit case of a weak segmentation is the detection of conspicuous points, see Fig. 9. Salient features may be covered by the generic equation:



where Λ stands for a local selection operation and operator h maximizes the saliency of the processed image field f(x). The area Tj over which the value of Fj is searched for is usually the whole image, although there would be no objection to concentrating on the center or top part of the image in search for specific events. As the information of the image is condensed into just a limited number of feature values, the information should be selected with precision for greatest saliency and proven robustness.

***Signs***: When one of the possible interpretations of an image is so preponderant that it can be considered the meaning of the image, the image holds a sign, characterized by the probability P on interpretation z:



with symbols as in (5). The analysis leads to a localization of a sign with its probability. Typical signs are an icon, a character, a traffic light, or a trademark. In the case of maps, the interpretation of map symbols and their spatial relationships provides access to the content of the map.

***Shape and object features:*** The theoretically best way to enhance object-specific information contained in images is by segmenting the object in the image. But, as discussed above, the brittleness of segmentation algorithms prevents the use of automatic segmentation in broad domains. In fact, in many cases, it is not necessary to know exactly where an object is in the image as long as one can identify the presence of the object by its unique characteristics. When the domain is narrow, a tailored segmentation algorithm may be needed more and, fortunately, also be better feasible. When segmentation is applied, we have:



where f(x) is the data field resulting from the processing above (equal to the image i(x) when g is the identity operator), sj is the segmentation operator for object j, and tj(x) indicates the object area Tj. For shape, Fj is a (possibly ordered) set of features from F for j:



where  represents an aggregation operation and h is the functional computing shape in this case. Object internal features are computed similar to (4). The object internal features are largely identical to the accumulative features, now computed over the object area.

***Description of structure and layout:*** When feature calculations are available for different entities in the image, they may be stored with a relationship between them, see Fig. 9 for an illustration. Such a structural feature set may contain feature values plus spatial relationships, a hierarchically ordered set of feature values, or relationships between point sets or object sets. The process is symbolized by:



where Tj,k indicates the kth part of the jth object and Hj,k is an (ordered) spatial relationship describing object j in k elements. Structural and layout feature descriptions are captured in a graph, hierarchy, or any other ordered set of feature values and their relationships.

**Interpretation and similarity:**

**Semantic Interpretation:** In content-based retrieval, it is useful to push the semantic interpretation of features derived from the image as far as one can.

*Semantic features aim at encoding interpretations of the image which may be relevant to the application.*

Of course, such interpretations are a subset of the possible interpretations of an image. To that end, consider a feature vector F derived from an image i. For given semantic interpretations z from the set of all interpretations Z, a learning phase leads to conditional probabilities:

****

A strong semantic feature with interpretation zj would generate a If the feature carries no semantics, it would generate a distribution independent of the value of the feature. In practice, many feature types will generate a probability distribution that is neither a pulse nor independent of the feature value. This means that the feature value “skews” the interpretation of the image, but does not determine it completely. Under the umbrella weak semantics, we collect the approaches that try to combine features in some semantically meaningful interpretation. Weak semantics aims at encoding, in a simple and approximate way, a subset of the possible interpretations of an image that are of interest in a given application.

***Similarity between Features:*** A different road to assigning a meaning to an observed feature set, is to compare a pair of observations by a similarity function. While searching for a query image iq(x) among the elements of the data set of images, id(x), knowledge of the domain will be expressed by formulating a similarity measure Sq,d between the images q and d on the basis of some feature set. The similarity measure depends on the type of features, see Fig. 10. The similarity of two feature vectors F, accumulative or object features alike, is given by:



At its best use, the similarity measure can be manipulated to represent different semantic contents; images are then grouped by similarity in such a way that close images are similar with respect to use and purpose. There is surprisingly little work dedicated to characterizing similarity measures. A few ideas, however, have emerged. A common assumption is that the similarity between two feature vectors F can be expressed as:



where g is a positive, monotonically non-increasing function and d is a distance function on F. This assumption is consistent with a class of psychological models of human similarity perception and requires that the feature space be metric. If the feature space is a vector space, d often is a simple Euclidean distance, although there is indication that more complex distance measures might be necessary. This similarity model was well-suited for early query by example systems in which images were ordered by similarity with one example.

A different view sees similarity as an essentially probabilistic concept. This view is rooted in the psychological literature [8] and, in the context of content-based retrieval, it has been proposed, for example, in [114]. A general form of such a similarity measure would be



Measuring the distance between histograms has been an active line of research since the early years of content-based retrieval, where histograms can be seen as a set of ordered features:



In content-based retrieval, histograms have mostly been used in conjunction with color features, but there is nothing against being used in texture or local geometric properties.

***Similarity of object Silhouettes:***



For shape comparison, the authors make a distinction between transforms, moments, deformation matching, scale space matching, and dissimilarity measurement. Difficulties for shape matching based on global transforms are the inexplicability of the result and the brittleness for small deviations. Moments, specifically their invariant combinations, have been frequently used in retrieval. Matching a query and an object in the data file can be done along the ordered set of eigen shapes or with elastic matching. Scale space matching is based on progressively simplifying the contour by smoothing. By comparing the signature of annihilated zero crossings of the curvature, two shapes are matched in a scale and rotation invariant fashion. When based on a metric, dissimilarity measures will render an ordered range of deviations suited for a predictable interpretation.

***Similarity of Structural Features:***



Bayesian framework is developed for the matching of relational attributed graphs by discrete relaxation. This is applied to line patterns from aerial photographs. A metric for the comparison of two topological arrangements of named parts, applied to medical images. The distance is derived from the number of edit steps needed to nullify the difference in the Voronoi diagrams of two images. Hierarchically ordered trees are compared for the purpose of retrieval by rewriting them into strings. A distance-based similarity measure establishes the similarity scores between corresponding leaves in the trees. At the level of trees, the total similarity score of corresponding branches is taken as the measure for (sub) tree-similarity. From a small size experiment, it is concluded that hierarchically ordered feature sets are more efficient than plain feature sets, with projected computational shortcuts for larger data sets.

***Similarity of Salient Features:*** Salient features are used to capture the information in the image in a limited number of salient points. Similarity between images can then be checked in several different ways. In the first place, the color, texture, or local shape characteristics may be used to identify the salient points of the data as identical to the salient points of the query.



where Fq and Fd are feature vectors of salient properties and g is an optional monotone function. A measure of similarity between the feature values measured of the blobs resulting from weak segmentation consists of a Mahalanobis distance between the feature vector composed of the color, texture, position, area, eccentricity, and direction of the two ellipses. If the features of the ellipse are collected in a vector F, the distance between q and d is given by



In the second place, one can store all salient points from one image in a histogram on the basis of a few characteristics, such as color on the inside versus color on the outside. The similarity is then based on the group-wise presence of enough similar points.



where Fq and Fd are histograms merely indicating the presence of salient points. The metric d(Fq,Fd) is now aiming at measuring the presence of the same set of salient points. Comparing sparsely occupied histograms has long been used in text retrieval, where vector space modeling implies the registration in a N-dimensional histogram F with as many dimensions as there are different words in the dictionary, typically 10,000. In a binary vector space, each dimension is expressing whether that word is present or absent in the text. A text is a point in this high dimensional space. Differences between the text d in the data file and the query q boil down to the intersection distance discussed above: distance



over all dimensions. The same strategy is used when comparing salient point features derived from different images. The intersection is appropriate when both q and d may be partially occluded in the image or cluttered with the background. When q is neither cluttered nor occluded but d may still be, the intersection should be replaced by the <- operation. Following the development of the vector space model in text retrieval, a weight per dimension may favor the appearance of some salient features over another. A third alternative for similarity of salient points is to concentrate only on the spatial relationships among the salient points sets Pq and Pd:



***Similarity at semantic level:*** knowledge-based type abstraction hierarchies are used to access image data based on context and a user profile, generated automatically from cluster analysis of the database.



The purpose of the bi-referential measure is to find all regions that are similar to two specified query points, an idea that generalizes to similarity queries given multiple examples.

***Learning an interpretation:*** As data sets grow large and the available processing power matches that growth, the opportunity arises to learn from experience. Rather than designing, implementing, and testing an algorithm to detect the visual characteristics for each different semantic term, it becomes possible to learn the semantics of objects from their appearance. A variety of techniques are discussed treating retrieval as a classification problem. One approach is principal component analysis over a stack of images taken from the same class z of objects. This can be done in feature space or at the level of the entire image. The analysis yields a set of “eigenface” images, capturing the common characteristics of a face without the need of a geometric model.

**Extensions to 1D and 3D cases:**

**CBIR as a universal approach to many tasks of pattern recognition:**