A Survey of Indexing and Retrieval of Multimodal Documents: Text and Images

Technical Report 2006-505

Nawei Chen
chenn@cs.queensu.ca
School of Computing
Queen's University
Kingston, Ontario, Canada

February 2006

Abstract

A document conveys information using multiple modalities, including text, layout/style and images. For example, journal articles usually have figures to illustrate experimental results, and the title in a journal article usually has a different font size than the body text. Indexing and retrieval using only text is the traditional way of IR (Information Retrieval). With the development of the Internet and Digital Libraries, it becomes increasingly important to develop IR techniques for intelligent indexing and retrieval of multimodal documents, such as web pages in HTML or XML format, scientific publications in PDF format and document images from scanned papers. In this paper, I make a survey of multimodal IR systems that combine the text and image modalities. Indexing and retrieval are two important components of an IR system. Given a collection of documents, indexing describes documents using an index language. Retrieval uses the results of indexing and finds related documents corresponding to a user's query. Text and image modalities use different indexing and retrieval techniques. Single-modality IR, either using text or images, has limitations. Multimodal IR aims to overcome the limitations in each single modality by combining them. The following issues in multimodal IR are addressed: various techniques to combine text and images; techniques to find relationships between text and images; noise and uncertainties in IR systems; and techniques to improve effectiveness of IR, such as Latent Semantic Indexing, user's relevance feedback, semantic network, and document clustering and classification.

Keywords: multimodal Information Retrieval, multimodal documents, document image retrieval, CBIR, text retrieval

Table of Content

1. Introduction 4
   1.1. Multimodal documents 4
   1.2. Overview of IR systems 5
   1.3. Issues 9
2. Indexing techniques 9
   2.1. Textual content indexing 9
      2.1.1. Noise-free text indexing 10
      2.1.2. Finding collateral text 10
      2.1.3. Indexing textual content from document images 11
   2.2. Visual content indexing 11
      2.2.1. Visual content indexing in CBIR 11
      2.2.2. Visual content indexing of document images 13
3. Retrieval models 14
   3.1. Text retrieval models 14
      3.1.1. The vector space model 14
      3.1.2. Probabilistic models 15
   3.2. CBIR retrieval models 15
      3.2.1. The vector space model 16
      3.2.2. Probabilistic models 16
      3.2.3. Other retrieval models 16
4. Combination of text and images 17
   4.1. Advantages and disadvantages of text and image modalities 17
      4.1.1. Text modality 17
      4.1.2. Image modality 17
   4.2. Overview of multimodal IR systems 18
   4.3. Combination techniques 20
      4.3.1. Sequential use of the text and image modalities 21
      4.3.2. Both modalities are used simultaneously 21
      4.3.3. Integration in a probabilistic model 23
5. Techniques to improve effectiveness of IR 24
   5.1. Latent Semantic Indexing (LSI) 24
      5.1.1. Introduction to Latent Semantic Indexing 24
      5.1.2. LSI in multimodal IR 24
   5.2. Semantic networks 25
      5.2.1. Introduction to semantic networks 25
      5.2.2. Semantic networks using both the text and image modalities 25
   5.3. User’s relevance feedback 28
      5.3.1. Introduction to relevance feedback 28
      5.3.2. Relevance feedback in multimodal retrieval 28
   5.4. Document clustering and classification 29
6. Conclusion 30
7. Acknowledgements 33
Reference 33

1. Introduction

Information Retrieval (IR) has different scopes of definitions. Broadly, IR is defined as “the field concerned with the structure, analysis, organization, storage, searching and retrieval of information” [Salt89]. Narrowly, IR refers to *ad hoc retrieval*, which is the task of “searching a collection of documents for relevant documents given an information need” [YaNe99]. In ad hoc retrieval, no prior knowledge is available on the relevance of documents. The system can only use the description of the information needs, which is often called a *query*. Typical examples of ad hoc retrieval systems are the search engines on the web, e.g. Google (www.google.com). In this survey, IR represents ad hoc retrieval.

Indexing and retrieval are two important components of an Information Retrieval system. Given a collection of documents, *indexing* describes documents using an index language [Rijs79]. *Retrieval* uses the results of indexing and finds related documents corresponding to a user’s query. In this survey, we focus on indexing and retrieval techniques for multimodal documents. In this section, first we introduce multimodal documents, and then we provide an overview of Information Retrieval systems. In the end, some issues are raised to guide our survey.

1.1. Multimodal documents

A document conveys information using multiple modalities, including text, layout/style and images. For example, journal articles usually have figures to illustrate experimental results, and the title in a journal article usually has a different font size than the body text. Examples of multimodal documents are shown in Figure 1. Indexing and retrieval using only text is the traditional way of IR. With the development of the Internet and Digital libraries, it becomes increasingly important to develop IR techniques for intelligent indexing and retrieval of multimodal documents.

Figure 1. Examples of multimodal documents

There is a great diversity of computer-accessible multimodal documents. Based on the source of creation, there are two major categories, *electronic documents* and *document images*. *Electronic documents* are generated from computational document editing tools. Examples include web pages in HTML or XML format, and scientific publications in PDF format. *Document images* (also called document page images) are scanned from paper documents or from fax machine, and are represented in image format (TIFF, JPEG, etc). A document image corresponds to a single page in a multi-page document. We will survey IR techniques for both types of multimodal documents.

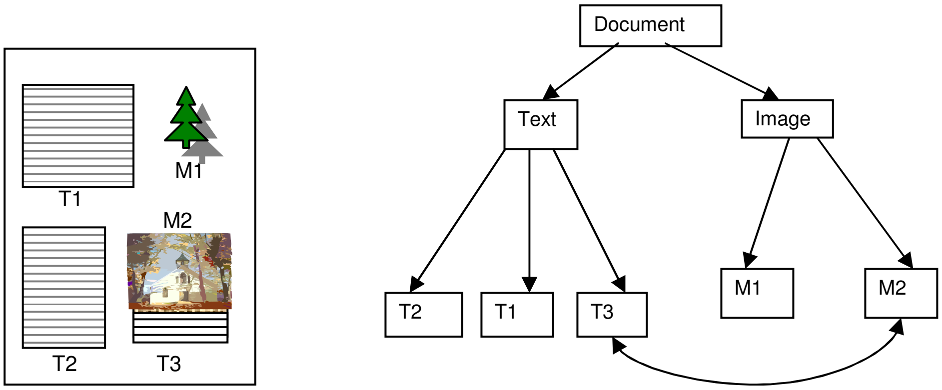


Figure 2. A multimodal document model

We define a multimodal document model as a set of multiple objects with two object types: text (T) or image (M). A text object represents a block of text. An image object represents an image. Images include graphical images (e.g. charts, graphs, maps, diagrams) and natural images (e.g. color or gray-scale photographs). Some text blocks have no association with images; these are called *independent text*. Other text blocks are associated with images, such as captions, or a set of keywords annotated to the images; these are called *collateral text* in this survey. Some images may not have collateral text. Figure 2 shows an example of a multimodal document. Text block T3 is related to the image M2. T1 and T2 are independent text blocks. Image M1 doesn't have collateral text. The right figure shows the hierarchical structure of the example multimodal document and the relationship between objects. Samples in Figure 1 can be represented in such a model. Most of the multimodal document models in our survey are simplified versions of the model in Figure 2, with each multimodal document containing a single image and possibly a collateral text block.

Unlike in electronic documents, there are no explicit objects in document images, so processing of document images relies on document image analysis techniques to find the text and image objects. A simplified diagram of document image analysis is shown in Figure 3. At first, text and images are separated from the document image through page segmentation and layout analysis. In this stage, collateral text may be associated with images. To get textual content, Optical Character Recognition (OCR) is applied to the text blocks. Noise and errors may occur in document image analysis. These have to be considered in subsequent indexing and retrieval.

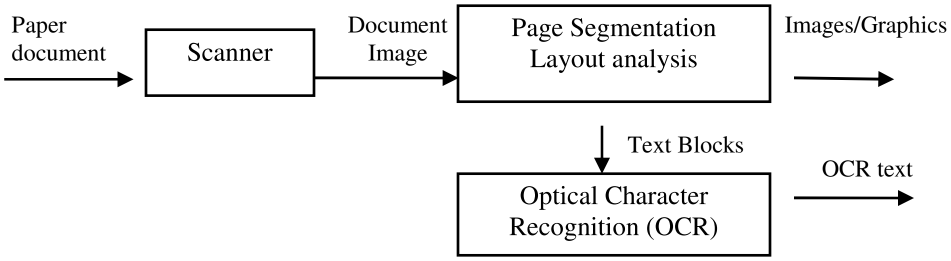


Figure 3. Procedures of document image analysis.

1.2. Overview of IR systems

Traditionally, a single modality, either text or images, has been used to retrieve content in multimodal documents. A simplified diagrammatic view of a single-modality IR system is shown in Figure 4. In general, there are four components in an IR system:

- **Indexing**: Indexing uses characteristic features to represent documents. Different features are extracted from either textual content of text blocks or visual content of images in a document depending on which modality is used. The indexing here refers to automatic indexing, i.e. indexes are automatically built without human intervention. The ideal indexing is to dynamically choose a set of features to represent documents given user's information needs. Automatic indexing techniques are surveyed in Section 2.
- **Query Formulating and Analyzing**: A user formulates a query through the query interface provided by the system. The system analyzes the query and represents it in the same internal format as used for document representations. Different systems differ in their friendliness and complexity of query interfaces depending on which modality is used for retrieval. A user query may be formulated in different ways. The query interface is important for users to form queries to represent their information needs. The details of query formulation and query interface are outside the scope of this paper.
- **Retrieval**: The system compares document representations and a query representation to retrieve documents using various retrieval models. Retrieval models are surveyed in Section 3. The result of a search is a set of *hits* containing both relevant (positive) documents and irrelevant (negative) documents.
- **Performance evaluation**: Precision and recall are the two most popular metrics to evaluate the effectiveness of text retrieval. *Precision* is the proportion of retrieved documents that are relevant, and *recall* is the proportion of relevant documents that are retrieved. Image retrieval and multimodal IR borrow these two terms for effectiveness evaluation. Techniques used for performance evaluation and benchmarking in IR are outside the scope of this paper.

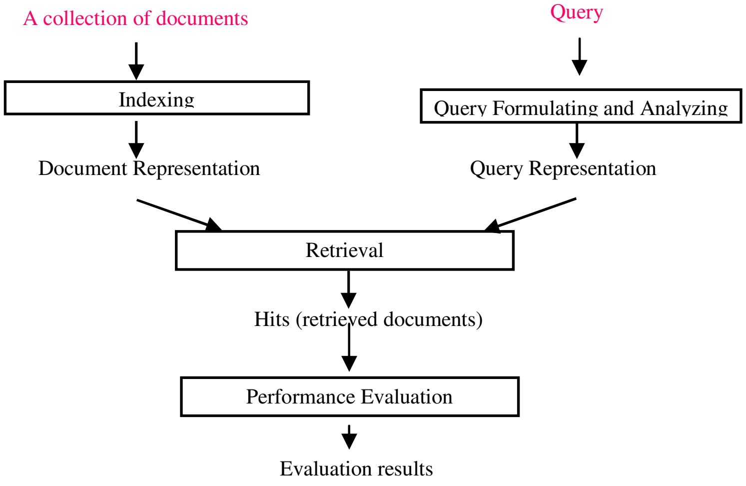


Figure 4. Diagrammatic view of a simplified single-modality IR system.

In the rest of this paper, we distinguish five types of IR systems: text retrieval, text-based image retrieval, CBIR (Content-Based Image Retrieval), document image retrieval and multimodal IR. They are briefly introduced in the following.

Text retrieval is the oldest branch of IR and it is well researched [Rijs79] [Salt89] [YaNe99] [Witt98]. Text retrieval only deals with text blocks, e.g. T1, T2, or T3 in Figure 2. A user represents a query using either a sequence of words or a Boolean combination *(AND, OR and NOT)* of words. For example, a user might input "*((image OR graphics) AND retrieval)*" or "image graphics retrieval" to find the information on image/graphics retrieval depending on the implemented retrieval model in the system. The indexing techniques are surveyed in Section 2.1.1 and retrieval models are surveyed in Section 3.1.

Text-based image retrieval and CBIR are two major approaches to retrieve images [Good00]. Text-based image retrieval is cross-modality retrieval, i.e. using collateral text to retrieve images. An example is to use collateral text block T3 to retrieve image M2 in Figure 2. The user represents a query using a textual string, and collateral text blocks are indexed. Collateral text blocks are linked to the image. There are uncertainties in finding collateral text blocks (discussed in Section 2.1.2). The retrieval techniques are similar to text retrieval or standard database retrieval using SQL [RuHC97]. We won't survey standard database retrieval techniques, which mostly deal with structured data.

CBIR deals with only images, e.g. M1 or M2 in Figure 2. CBIR uses visual content extracted directly from images for indexing and retrieving. The query may be an example image, a user's sketch or image attributes. An example of using an example image as a query is shown in Figure 5. An example of using a sketch or specifying color attributes as a query is shown in Figure 6. CBIR has drawn a lot of attention since the early 90's. There are several comprehensive surveys on CBIR [RuHC97] [RuHu99] [YoIc99] [SmWS00] [AnKJ02] [VeTa02]. Commercial CBIR systems are available, such as IBM's QBIC. Most of the systems surveyed in this paper are experimental systems in academia. Visual content indexing techniques are surveyed in Section 2.2. CBIR retrieval models are surveyed in Section 3.2.

Document image retrieval uses the results from page segmentation and/or document layout analysis (shown in Figure 3). After a preprocessing step to separate text from images, most document image retrieval systems deal with either text or images. There are comprehensive surveys on indexing and retrieval techniques for document images [Doer98] [MiCh00]. Textual content indexing and retrieval is similar to the techniques in text retrieval. However, errors in OCR must be considered. Basic textual content indexing methods are surveyed in Section 2.1.3. Visual content indexing is similar to the indexing in CBIR. The difference is that visual content may come from images within a document image or from the whole page image.

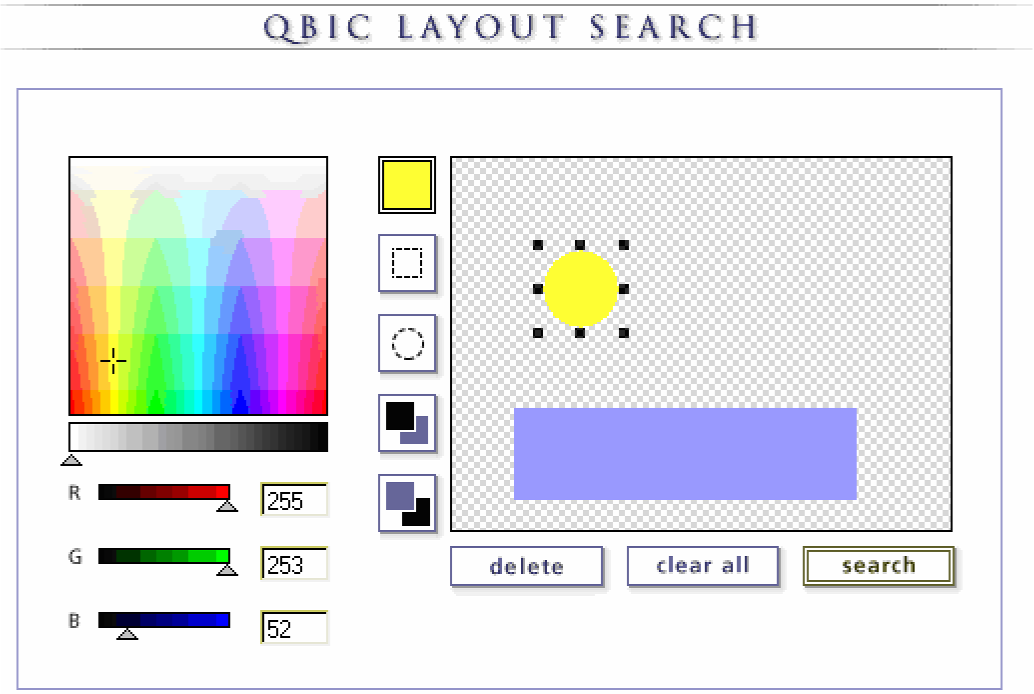
Multimodal IR systems use both the text and image objects. IR using either the text or image modality has advantages and limitations (surveyed in Section 4.1). The combination of text with images is demonstrated to improve retrieval performance [ScCS99] [BaFo01]. As far as we know, there is no existing survey of this field. Multimodal IR systems are based on single-modality IR systems and they are more complex than single-modality IR systems. Composite queries including both text and images may be used. Both textual and visual content are indexed and various combination techniques are developed to combine retrieval results using both text and images. We will survey combination of text and images in Section 4.

DATABASE TYPE: Photographs

MATCHING TYPE: Fuzzy

FEATURES: Color invariant

Figure 5. Example of using a sample image as a query in the PicToSeek CBIR system. (screen shot from http://zomax.wins.uva.nl:5345/ret_user/).



R 255

G 253

B 52

Figure 6. Example of the user interface in QBIC (screen shot from wwwqbic.almaden.ibm.com). This figure is better viewed in color. Users search images by selecting colors from a palette or by sketching shapes on a canvas.

1.3. Issues

Multimodal IR is a large research area. It is closely related to the following research areas: text retrieval, image retrieval, document image retrieval, and document clustering/classification. Each of these research areas is rather large and actively researched, so it is impossible to cover all the state-of-the-art. In this survey, we focus on the following issues:

- The basic indexing techniques and retrieval models for the text and image modalities in multimodal documents. The indexing techniques and retrieval models all aim to provide an effective and efficient IR.
- Adaptable and reconfigurable IR. Techniques in text retrieval have been adapted to image retrieval, or multimodal IR. State-of-the-art methods in other research areas may be adapted to multimodal IR. For multimodal IR, it is challenging to develop a general approach to combine text and images, which may be adapted to different types of documents or different modalities.
- Combination of text and images in multimodal IR. Retrieval using either the text or image modality has advantages and disadvantages. These modalities may be combined to overcome the limitations in a single-modality IR system.
- The relationship between collateral text and image. When describing image content, collateral text and image complement each other. There are also redundancies between text and images. One modality may help make the other modality more informative and more precise.
- Noise and uncertainties in IR systems. Uncertainties exist in various parts of IR systems. There are uncertainties when representing a user's information needs as a query, when indexing either collateral text or text from OCR, and when matching a query with documents. These uncertainties are dealt with in various ways.
- Techniques to improve effectiveness of IR, such as Latent Semantic Indexing, user's relevance feedback, semantic network, and document clustering and classification. These techniques are surveyed in Section 5.

2. Indexing techniques

Both textual content and visual content may be indexed in multimodal documents. Section 2.1 surveys textual content indexing, which usually uses a sequence of index terms to represent text. Various types of image features might be extracted to represent the visual content. Visual content indexing is surveyed in Section 2.2.

2.1. Textual content indexing

Text in an electronic document is assumed noise-free, in contrast with text in document images, where OCR has errors. Section 2.1.1 summarizes noise-free text indexing used in text retrieval. Indexing of collateral text used in text-based image retrieval is similar to noise-free text indexing. However, finding collateral text is not a trivial task in some documents. Techniques to find collateral text are reviewed in Section 2.1.2. Section 2.1.3 summarizes special techniques that are developed to index textual content in document images.

2.1.1. Noise-free text indexing

Index terms (or *keywords*) are the basic units to represent textual content of documents. In English natural language text, index terms are obtained using the following procedures [YaNe99]:

1. (1) Lexical analysis of the text is performed to extract candidate words to be used as index terms from the documents. During lexical analysis, various issues arise, such as how to deal with digits (*CISC 999*), punctuations (*depth.doc*), hyphens (*on-line*), etc.
2. (2) Optionally, stop words are removed. Stop words are words that occur too often to hold information, such as *the, of, in*. The removal of stop words usually improves precision, and reduces the size of the indexing structure considerably. It might reduce recall.
3. (3) Optionally, words are stemmed. Different word forms may bear similar meanings (e.g. *search, searching*). Stemming aims to group words with a common stem together. Stemming usually improves recall, and reduces the size of the indexing structure. In some cases it might reduce precision.
4. (4) Optionally, multi-word phrases are identified as additional indexing terms.
5. (5) Optionally, syntactic analysis of a sentence is performed and words from various grammatical classes are chosen, such as nouns, verbs. Usually nouns are chosen since they are assumed to capture most of the semantics of a document.

There is no conclusive evidence that these optional text processing steps yield consistent improvement in retrieval performance, especially when the collection of documents is general and heterogeneous [YaNe99]. That is the reason why some web search engines index all the words.

Indexing constructs an *inverted index* of word-to-document pointers. The inverted file is designed for rapid merging of document lists and calculation of similarities. We won't survey the data structures used for indexing.

2.1.2. Finding collateral text

Collateral text indexing is similar to noise-free text indexing. However, there are uncertainties when locating collateral text in some multimodal documents, e.g. web pages. We briefly address this issue since many multimodal IR systems in our survey retrieve images from web pages. Unlike journal articles, most images on a web page do not have an explicit caption. In web pages, the collateral text of an image may be extracted from many sources, such as the page title, the image file name, the alternate text and the surrounding text which include top, bottom, left and right directions [SwFA97].

The collateral text may be incomplete or irrelevant. If the surrounding text around an image is not considered (e.g. only titles in a news archive are used [ZhGr02]), the collateral text may be incomplete. On the other side, if using the full-text of the web page as the collateral text, some text may be irrelevant. Various techniques are developed to find surrounding text related to an image. For example, Mukherjea and Chost use criteria like visual and syntactic distance between images and potential captions to relate text to images [MuCh99].

Since the relevancy or importance of various text sources can be different, some systems give different weights to the text from different sources. In ImageRover, a word relevant to an image is identified based on its frequency of occurrence in the HTML text, its position with respect to the image, and its style [ScCS99]. Words with specific HTML tags are given higher weights. How to assign suitable relevance weights to different sources is a difficult problem.

2.1.3. Indexing textual content from document images

Due to the errors in OCR (Optical Character Recognition), robust indexing techniques are needed to index textual content from document images. Indexing of document images has been comprehensively surveyed in [Doer98] [MiCh00]. There are basically two indexing approaches:

- **Index textual content using OCR results**

  This type of indexing requires a full conversion of the text in document images using OCR. Many techniques have been proposed to deal with OCR errors. Lopresti uses an NGram method to index OCR text [Lopr96]. NGrams split up a document into n-character terms. For example, "DOCUMENT" is split up into several trigrams: DOC, OCU, CUM, UME, MEN, and ENT. Each trigram is an indexing term. This NGrams indexing produces a large size of indexing structure. Another approach is to correct OCR errors using a dictionary or language statistics, and then noise-free text indexing techniques are used [TaBC94]. This approach usually can't make a total correction of OCR errors.

- **Index images of text without OCR**

  OCR is error-prone for poor quality document imags (e.g. fax or documents from copier machine, historic documents, and handwriting). Textual content may be indexed directly from images without OCR. For example, Character Shape Codes based on character image features are used to index the textual content of the documents [SmSp97]. In the indexing procedure, first image segmentation is performed at word and character levels. Then each character in a word is mapped to a Character Shape Code without being identified. These codes together form a Word Shape Token for each word, which works as an index term. This approach can be cheaper than a full OCR conversion and may be more robust to noise. Also it works for all languages and scripts. It is unsuitable to deal with words with touching characters since characters can't be segmented. Alternatively, word image indexing may be used [LuZT04]. The image-level features of an entire word are coded without character segmentation and OCR.

2.2. Visual content indexing

The goal of visual content indexing is to extract characteristic image features and organize them in a way to facilitate CBIR. Multiple features may be extracted from an image and may be represented in multiple formats [RuHu99]. Image feature extraction borrows the research achievements from pattern recognition and computer vision. It is impossible to review all the state-of-the-art image feature extraction techniques. In Section 2.2.1, I summarize the most commonly used image features in CBIR. Some specific features, such as those used for medical images, human faces and finger prints, need domain-specific knowledge and they are covered elsewhere in the pattern recognition and computer vision literature. In Second 2.2.2, I survey special visual content indexing techniques for document images.

2.2.1. Visual content indexing in CBIR

Image features can be extracted at a global level, i.e. from a whole image, or a local level, i.e. from regions or objects of an image. To obtain the local image features, an image is often divided into parts first [RuHu99]. The simplest way is to partition an image into data independent parts. This kind of partition does not generate perceptually meaningful regions but is a way of representing the global features of the image at a finer division. An alternative method is to segment the image into salient regions according to some criterion, such as color and texture [CaTB99]. Region segmentation is not a trivial task. Unavoidably, there are uncertainties in region segmentation.

Color, texture, shape, and spatial layout features are the most commonly used visual features in CBIR. In the following, I provide an overview of these features.

Color

Color features are commonly used for retrieval of color images. They are easy to compute and are insensitive to small changes in viewing positions. Color features extracted from various color spaces (e.g. HSV, LUV) take various formats. Here, color histogram, color moments, and color correlogram will be discussed. A color histogram represents the distribution of the number of pixels for each quantized bin in each color channel. A normalized color histogram is used in ImageRover [ScTC97]. Stricker and Orengo use color moments to characterize the color distribution [StOr95]. Up to third order color moments, i.e. mean, variance and skewness, are extracted. A color correlogram is a three-dimensional histogram, of which the first and the second dimensions are the colors of any pixel pair and the third dimension is their spatial distance [HuKM97]. It is used to characterize not only the color distributions of pixels, but also the spatial correlation of pairs of colors.

Texture

Various texture features have been effectively used in CBIR systems combined with other features, such as in QBIC [Flic95] and in MARS [HuMR96]. The commonly used texture measures are: Haralick's gray level co-occurrence features, Tamura texture features and wavelet transform. Haralick's co-occurrence features represent the gray level spatial dependence of texture [HaSD73]. The co-occurrence matrix is based on pixel values at various orientation and distances. Statistics computed from the matrix include *contrast*, *inverse difference moment*, and *entropy*. The Tamura texture features include *coarseness*, *contrast*, *directionality, line-likeness, regularity,* and *roughness* [TaMY78].The Wavelet transform [Mall89] provides a multi-resolution approach to texture analysis and classification.

Shape

Shape features are extracted from an object or a region after image segmentation. Since robust and accurate image segmentation is difficult to obtain, shape features are not as commonly used as color and texture for a general collection of images. They have been limited to special applications where objects or regions are readily available, such as retrieval of line drawings [LoMo95] and retrieval of trademark images [JaVa98]. Mehtre et al compared various shape representations for CBIR [MeKL97]. In general, the shape representations can be divided into two categories: boundary-based and region-based. Boundary-based features use only the outer boundary of the shape. The most successful boundary-based features are Fourier descriptors. The main idea of a Fourier descriptor is to use the Fourier transformed boundary as the shape feature [PeFu77]. Region-based features use the interior of the shape region. The basic region-based features are centroid, area, eccentricity, circularity, and statistical moments.

Spatial layout

Spatial layout encodes the absolute or relative position of segmented objects or regions, for example, a blue region (e.g. sky) is on the top of a green region (e.g. grass). Spatial attributes can be represented in different ways, such as 2D-strings [ChSY87], spatial quad-tree [Same84], and symbolic images [GuRa95]. 2D-strings encode the spatial relationships of objects in horizontal and vertical directions using two strings. Symbolic images represent the original images logically by uniquely labeling image objects with symbolic names. Spatial relationships in a symbolic image are represented as edges in a weighted graph. Due to the difficulty of accurate segmentation, spatial features of segmented objects in an image are also used in limited applications.

Here we finish the overview of most commonly indexed image features in CBIR. These features are typically calculated off-line and stored for each image. There is no single optimal feature set for a large collection of heterogeneous images. Usually multiple types of features are extracted from the images. The critical problem and the open issue is how to dynamically choose a best set of robust features [RuHC97]. Goodrum points out that the early developments of CBIR systems focus primarily on the use of features that can be computationally acquired, but little has been done to identify the visual attributes suited for various tasks and collections [Good00]. For efficient retrieval in a large set of images, internal indexing structures for the feature data are also important. A survey of indexing structure is beyond the scope of this work.

2.2.2. Visual content indexing of document images

In document images, visual content may come from images within a document image. Indexing of images within a document is the same as visual content indexing in CBIR. A difference is that images in most CBIR systems are natural images in color (an example is Figure 5 in Section 1), so the color features are extensively researched. In contrast, most document images are either black-and-white or gray-scale images, so a different set of features are used.

A whole document image may be treated as a single image and indexed for retrieval based on visual similarity. Mostly-text document images have more distinctive structural characteristics than the natural or graphical images in CBIR. Figure 7 shows examples of mostly-text document images with distinctive structural features. Structural features are extracted from the results of document layout analysis. For example, a modified XY-tree representation is used for document page retrieval [CeMS02]. The XY-tree representation is a well-known approach for describing the physical layout of documents [NaSe84]. The root of an XY-tree is associated with the whole document image. The document is split into regions that are separated by white spaces. Horizontal and vertical cuts are alternately performed. Each tree node is associated with a document region.

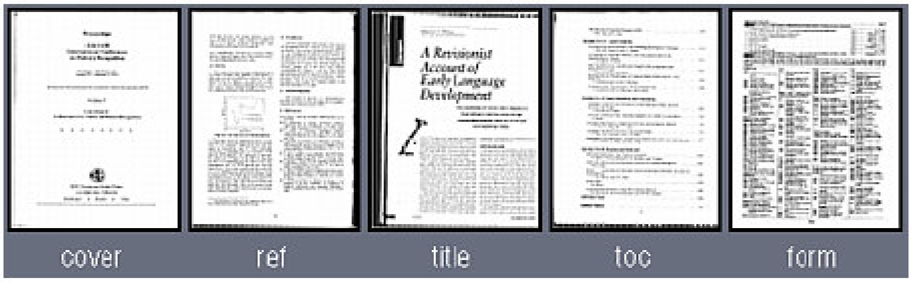


Figure 7.Examples of different types of document page images (cover, reference, title, table of contents and form) with distinctive structural features, reproduced from [ShDR01].

.

3. Retrieval models

A retrieval model describes how document representations and a query representation are compared to retrieve documents. In Section 3.1, we review text retrieval models. Text retrieval models are well-developed and they have been adapted to other branches of IR. In Section 3.2, we review CBIR retrieval models. We won't review the retrieval models in text-based image retrieval and document image retrieval since they are closely related to either text retrieval or CBIR retrieval. The text retrieval models may be enhanced to deal with noisy OCR text, such as approximate string matching. Multimodal retrieval models are a combination of text and CBIR retrieval models, and various combination techniques are developed, which will be surveyed in Section 4.

3.1. Text retrieval models

There are two major categories of text retrieval models: exact matching, e.g. Boolean model, and ranked retrieval, e.g. vector space model and probabilistic model. Exact matching matches the query with the documents. It assumes all matching documents are equally relevant to the query. It doesn't provide a scheme to rank documents in an order of estimated relevancy. Ranked retrieval models measure the degree of similarity between a query and a document, and return a set of documents ranked by how similar they are to the query. A vector space model ranks documents by geometric similarity to the query. A probabilistic model ranks documents by probabilistic relevance. They are not mutually exclusive. They provide similar solutions, but different interpretations for a same problem. Other models combine various retrieval methods. In the following, we introduce the vector space model and probabilistic model.

3.1.1. The vector space model

The vector space model [SaWY75] is the most popular text retrieval model. A collection of documents $D=(d_1, .., d_m)$ can be viewed as vectors in a very high dimensional vector space. Every term in the index language $(t_1, .., t_n)$ becomes an independent dimension (or a feature). A document d_i is represented as an n-dimensional vector $\overrightarrow{d_i} = (w_{1i},...,w_{ni})$. W_{ji} is the weight of t_j in d_i, expressing how well feature t_j describes the content of document d_i. The query is regarded as a short document and is represented as a vector in the same space as the documents. The retrieval function computes the similarity between the description vectors of all documents with the description vector of a query, and returns the *k* documents ranked by the closeness of their vectors to the query.

Various similarity functions are used, such as Jacaard, dot product, and cosine similarity [YaNe99]. Cosine similarity is the most common one. For normalized vectors, the cosine is simply the dot product as shown in eq. (1).

$$sim(d_i,d_k)=\vec{d}_i\cdot\vec{d}_k=\sum\nolimits_{j=1}^{n} w_{ji}w_{jk},\ \vec{d}_i \textit{ and } \vec{d}_k \textit{ are normalized document vector} \qquad (1)$$

The vector space model returns ranked documents in an order that the documents most likely to be useful to the searcher are listed first. It benefits from queries with many terms, since not all of the terms in a query need to match the documents.

Term weighting

Proper weighting of terms affects the effectiveness of the vector space model. The index terms in a collection of documents have different frequency. There are three main factors when weighting terms: (1) *tf* (term frequency), the frequency of the term in a document. Terms that are frequent in a document are important indicators of the document's content. (2) *df* (document frequency), the number of documents containing the term. Terms that are frequent throughout the collection are not useful in distinguishing a document's content. Usually, *idf* (inverse document frequency) is used to represent the importance of a term. (3) Document length. Documents usually vary in lengths. A longer document tends to score higher because it uses more terms, and the same words may be used repeatedly.

Various methods for weighting terms have been developed. The most popular one is $tf \cdot idf$ weighting. $tf \cdot idf$ weighting has many variations. A simple $tf \cdot idf$ weighting is shown in eq. 2. The weights are usually normalized to discount the effects of document length.

$$tf \cdot \ln \frac{no.\ of\ documents}{df} \qquad (2)$$

3.1.2. Probabilistic models

Probabilities provide a principled foundation for uncertain reasoning. According to the *Probability Ranking Principle*, if documents are treated independently, the optimal ordering of returned documents is by decreasing probability of relevance [Rijs79]. The objective of a probabilistic model is to compute the probability that a document (D) is relevant given a query (*q*), i.e. *P(D is relevant| q).* Various probabilistic models have been surveyed in [CrLR98]. They differ in how they evaluate the probability of relevance.

Language models for retrieval

Since 1998, probabilistic language models have become increasingly popular in text retrieval [PoCr98] [Hiem98] [MiLS99] [LaZh01]. Each document is represented as a generative probabilistic model of terms. Each document model defines a probability distribution over the terms in the vocabulary. The model is estimated from a sample of text representative of that model. Usually a unigram estimate of words is used, i.e. the terms are independent given a particular language model. Query terms are assumed to be independent observations from a document model. Documents are ranked by the probability *P(D|q),* the probability of a relevance of a document model given a query. It is the same as ranked by *P(q|D)* (the probability of a query generated by a document model), according to Bayes Theorem $p(D|q) = \frac{p(q|D) * p(D)}{p(q)}$, in which *p(q)* is constant, and the prior probability *p(D)* is often treated the same for all documents. Languages models