Chapter 13 – Multimedia: Information Representation and Access

Suzanne Little, Evan Brown, Stefan Rüger
Knowledge Media Institute, The Open University
s.little@open.ac.uk

Introduction

The rich, multi-dimensional, multi-modal data channel offered by multimedia brings with it unique challenges for information retrieval. The old saying equates a picture to a thousand words. While this is true for the amount of information that a picture can convey to a user, from a data management and information retrieval perspective the equation is not so straightforward. More than a thousand words and more complex representations are often needed to successfully identify relevant media and assist users in their data finding missions.

Traditionally multimedia documents, both digital and analogue, have been indexed and queried using text descriptions in the form of metadata. Metadata for multimedia can be modelled or classified in a number of ways (e.g., Boll et al., 1998; Troncy et al., 2007). It is useful to group multimedia metadata into four rough categories, as shown in figure 1, arranged by the level of human effort potentially required to create the descriptors, to illustrate the concept of the "semantic gap". The semantic gap in multimedia information retrieval describes the difficulty in automatically generating usable high-level semantic descriptions (e.g., dog, party, ocean etc.) from automatically extracted low-level media features such as colour, texture, tempo, pitch etc. The concept originates in psychology (Finke, 1989) but its use within multimedia information retrieval is defined by Smeulders et al. (2000) as "the discrepancy between the information that one can extract from the visual data, and the interpretation that the same data has for a user".

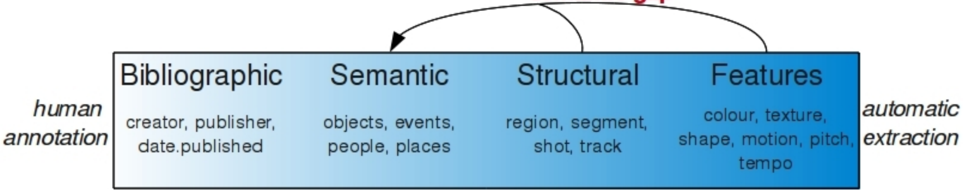

semantic gap
human annotation
Bibliographic
creator, publisher, date.published
Semantic
objects, events, people, places
Structural
region, segment, shot, track
Features
colour, texture, shape, motion, pitch, tempo
automatic extraction

Figure 1: Categories of multimedia descriptors

When managing high-volumes of multimedia data, it is very attractive to try and generate quality semantic annotations from features automatically extracted from the pixels or bits. Not only is this cheaper and more efficient than manual human annotations but it also improves the consistency of the annotations by reducing the influence of user opinion. Bridging the multimedia semantic gap is the target of a large field of research. We will discuss some aspects of it in section 1.2 in the context of automatically annotating media but good general starting points can be found in Smeulders et al. (2000), Zhao and Grosky (2001) and Hare et al. (2006b).

There is an alternative to indexing media documents with text descriptions and then searching using text queries. The concept of content-based media retrieval seeks to enable users to query media collections using media queries. Therefore a user may search for photographs that are similar to a given example or look for a particular song by humming part of it or find video based on some screenshots. This style of search bypasses the need to generate semantic descriptions from features and uses the similarity of the features themselves to assume semantic similarity. Content-based media retrieval will be discussed in more detail later.

Multimedia information retrieval can use metadata or descriptors from different points along the scale given in figure 1 to support different information needs, types of media or available descriptions. In this chapter we discuss the main classes of indexing and retrieval approaches for multimedia – metadata-driven, piggy-backed text, automated annotation, fingerprinting and content-based – and examine content-based retrieval in greater detail. As part of this we will look at the ways in which information in multimedia data can be represented and indexed to support information retrieval.

Metadata-driven Retrieval

The current best practice for indexing multimedia collections is via the generation of a library card, i.e., a dedicated database entry of metadata such as author, title, publication year and keywords. Depending on the concrete implementation, these can be found with SQL database queries, text-search engines or XML queries, but all these search modes are based on text descriptions of some form and are independent of the structure of the actual objects they refer to, be it books, CDs, videos, newspaper articles, paintings, sculptures, web pages, consumer products etc.

Metadata are pieces of information about a multimedia object that are not strictly necessary for working with it, but that are useful to

- *describe* resources so they can be indexed, classified, located, browsed and found
- *store technical* information, such as data formats and compression schemes
- *manage* resources such as their rights or where they are currently located
- *record preservation* actions
- *create usage* trails, e.g., which section of a video has been watched how many times

When using metadata to index collections of media (or indeed any other objects) it soon becomes clear that metadata must be able to be exchanged, read and indexed and therefore should adhere to agreed upon formats for its syntactic and semantic structures. Multimedia is increasingly described using structured documents in XML format. This allows for processing, harvesting, indexing and interchange by a wide variety of tools and systems.

Media are often described using common bibliographic metadata standards such as Dublin Core or MARC (Machine Readable Catalogue). These standards represent the spectrum of metadata complexity. Dublin Core (DC)[1] consists of 15 elements such as title, creator, subject, description, date – making it very easy for anyone to create descriptive metadata for their multimedia objects. In contrast, the MARC standard[2] consists of several hundred entries with complex rules and the

[1] http://dublincore.org/
[2] http://www.loc.gov/marc/

creation of records is only possible for trained specialists. However the result is a comprehensive description that is easily shared across libraries.

Bibliographic metadata generally has only superficial descriptions of the media content such as subject classifiers, keywords, titles or simple free text descriptions. To describe the content of media the most appropriate standard is MPEG-7. MPEG-7, the Motion Picture Experts Group Multimedia Content Description Standard, is a format for the description and search of audiovisual resources.

MPEG-7 descriptions cater for still images, graphics, 3d models, audio, speech, video, and information about how these elements are composed in a multimedia presentation. They care about the content of the multimedia object on various levels, from low-level machine-extractable features, to high-level human annotations, but they do not engage with the way the content is represented: physical world objects such as a drawing on paper can have an MPEG-7 description in the very same way as a compressed digital TIFF image.

As with other metadata standards, there is not a single “right” MPEG-7 file for a particular multimedia object. MPEG-7 allows, and encourages, different levels of granularity in the description depending on the application type. Although MPEG-7 puts great emphasis on content description, more traditional metadata such as media type, rights information, price and parental ratings can also be included. This is further encouraged through the use of “profiles” that define subsets for specific applications. The Digital Audiovisual Profile (DAVP) (Schallauer et al, 2006) is an example.

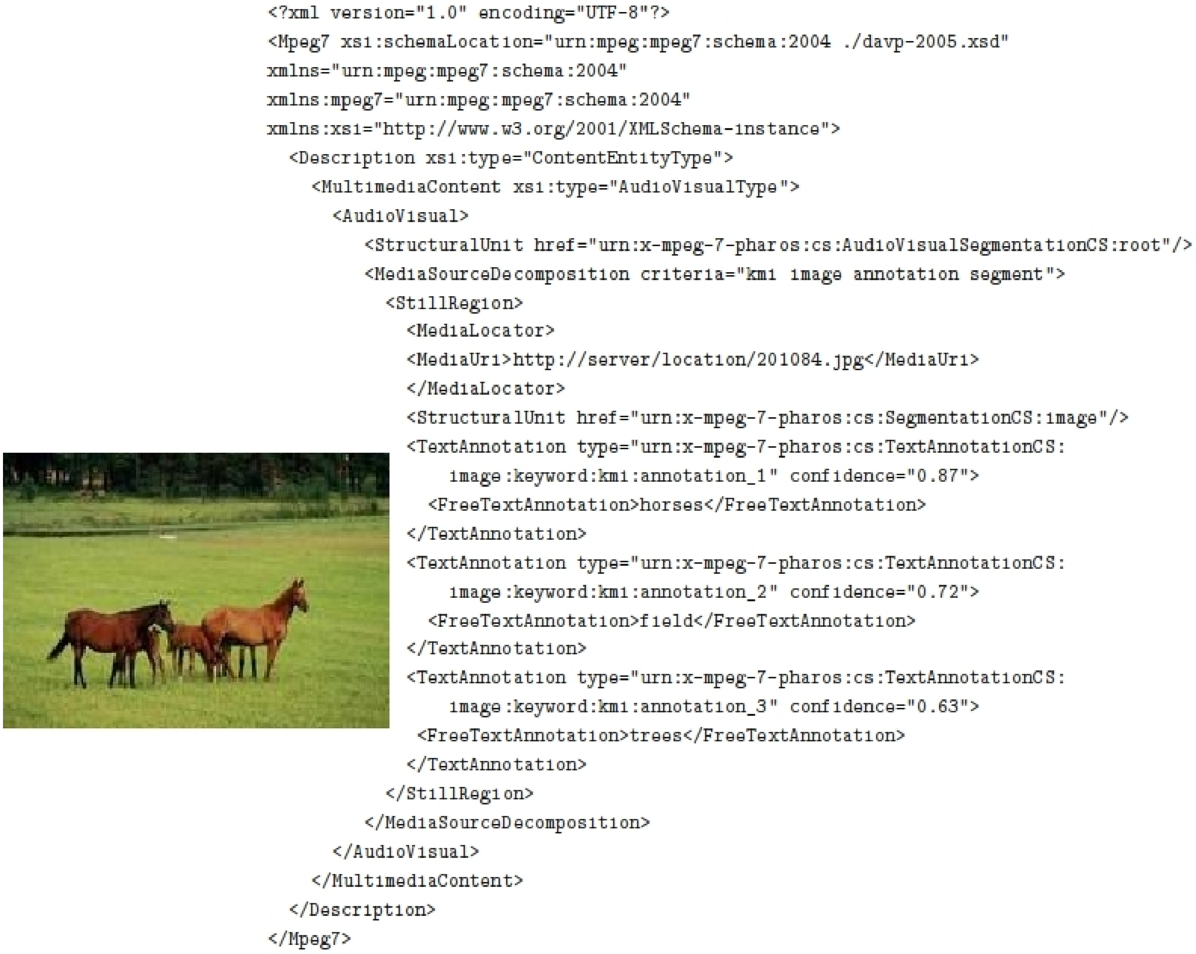
The three main elements of MPEG-7 are:

- *Descriptors* to define the syntax and the semantics of each feature, and *description schemes* to specify the relationships between their components, which in turn may be descriptors and description schemes
- *Description definition language* to define the syntax of the MPEG-7 description tools and to allow the creation of new description schemes and descriptors
- *System tools* and *reference implementations* to support binary coded representation for efficient storage and transmission, multiplexing (combination) of descriptions, synchronization of descriptions with content, management and protection of rights

Figure 2 shows an MPEG-7 encoding of the results of an algorithm that predicts the presence of tree, field and horses in a photograph with various levels of confidence generated as part of the PHAROS project (Bozzon et al. 2009). More information about MPEG-7 can be found at the MPEG[3] website and the MPEG-7 Consortium website[4] or from Nack (1999) and Manjunath (2002).

[3] http://www.chiariglione.org/mpeg

[4] http://mpeg7.nist.gov



```
<?xml version="1.0" encoding="UTF-8"?>
<Mpeg7 xsi:schemaLocation="urn:mpeg:mpeg7:schema:2004 ./davp-2005.xsd"
xmlns="urn:mpeg:mpeg7:schema:2004"
xmlns:mpeg7="urn:mpeg:mpeg7:schema:2004"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Description xsi:type="ContentEntityType">
    <MultimediaContent xsi:type="AudioVisualType">
      <AudioVisual>
         <StructuralUnit href="urn:x-mpeg-7-pharos:cs:AudioVisualSegmentationCS:root"/>
         <MediaSourceDecomposition criteria="kmi image annotation segment">
           <StillRegion>
             <MediaLocator>
             <MediaUri>http://server/location/201084.jpg</MediaUri>
             </MediaLocator>
             <StructuralUnit href="urn:x-mpeg-7-pharos:cs:SegmentationCS:image"/>
             <TextAnnotation type="urn:x-mpeg-7-pharos:cs:TextAnnotationCS:
                 image:keyword:kmi:annotation_1" confidence="0.87">
               <FreeTextAnnotation>horses</FreeTextAnnotation>
             </TextAnnotation>
             <TextAnnotation type="urn:x-mpeg-7-pharos:cs:TextAnnotationCS:
                 image:keyword:kmi:annotation_2" confidence="0.72">
               <FreeTextAnnotation>field</FreeTextAnnotation>
             </TextAnnotation>
             <TextAnnotation type="urn:x-mpeg-7-pharos:cs:TextAnnotationCS:
                 image:keyword:kmi:annotation_3" confidence="0.63">
              <FreeTextAnnotation>trees</FreeTextAnnotation>
             </TextAnnotation>
           </StillRegion>
         </MediaSourceDecomposition>
      </AudioVisual>
    </MultimediaContent>
  </Description>
</Mpeg7>
```

FIGURE: 2 MPEG-7 EXAMPLE FOR AUTOMATED TEXT ANNOTATION OF AN IMAGE

External descriptive metadata is not the only source of information about technical aspects of media objects. Much information is also recorded within file formats based on settings used during capture. For example the EXchangable Information Format (EXIF)[5] is an agreed upon method by camera manufacturers for incorporating technical capture information from the camera and storing in image file formats such as TIFF or JPEG. Other examples include XMP from Adobe[6] or the ID3 tagging system[7] for describing MP3 audio files. This information can be added at creation time, extracted automatically from the media file and used to index multimedia collections.

Witten et al (2009, Chapter 6) give a deeper insight into metadata in general and specifically its use within digital libraries, but see also (Lagoze and Payette, 2000; Zeng and Qin, 2008).

PIGGY-BACK TEXT AND AUTOMATIC ANNOTATION

Another method for indexing media using traditional text techniques (discussed in chapter 9) is to identify or generate supporting text describing the media content. This can be termed *piggy-back text* retrieval.

The most straightforward way to describe media content is to exploit any surrounding or associated text in the multimedia document. This is very commonly used in current web image search engines

[5] http://www.cipa.jp/english/hyoujunka/kikaku/pdf/DC-008-2010_E.pdf
[6] http://www.adobe.com/products/xmp/
[7] http://www.id3.org/

that extract text from the surrounding html to identify media content. It is also plausible for digitised traditional print media such as books or articles where the captions and text descriptions can be mined and indexed in the same ways as other textual descriptions. Text can also be generated more directly from the content by converting speech to text or performing OCR on image components or associated video subtitles (Jung, 2004; Zhang, 2008).

It is also possible to use a more literal text-based representation of the abstract media content. This is a common approach in music retrieval where musical "words" can be formed from pitch, timing, duration information contained in the notation. These words then act as surrogate text for music representation; query by humming can thus be treated as a text retrieval problem. Downie and Nelson (2000) were the first to map music to text in this way. Later Doraisamy (2005) deployed this principle and extended it to both polyphonic music, where more than one note is present, and rhythm, i.e., music retrieval by tapping.

The major advantage of indexing media by associating or generating text descriptions is the speed with which large collections can be indexed and the ability to exploit traditional methods for indexing and retrieval. For massive datasets or web-scale applications where human annotation is not feasible this is particularly attractive. The challenge is ensuring independent accuracy of the descriptions when they are heavily influenced by the context of the media. For example, a photograph of a luxury car used to illustrate a web article discussing consumer spending would not necessarily be indexed with the terms "car Porsche speed flashy" etc. as these words do not appear in close proximity to the image in the document. Automated annotation tries to overcome this limitation by using the media content independent of the context in which the media occurs to generate semantic text descriptions of media.

Automatic annotation uses training sets of previously labelled media to determine the probability of a keyword applying to a given, un-annotated, media item. In this way it attempts to automate the task of manually labelling media with subjects or keywords from a given set of terms. Features, such as the colour, shape or texture of specific parts of the image or the tempo, pitch, mode of the audio, are extracted to differentiate between media items. A variety of machine-learning and statistical approaches to automatic annotation have been proposed and researched. Good starting points include Smeulders et al. (2000), Lew et al., (2006) and Datta et al., (2008).

One large problem with automatic annotation is the need to create a training set of sufficient size, variety and accuracy with which to build the automatic annotator. Automatic annotation has demonstrated the most success in specific, narrow subjects with a limited vocabulary and well-controlled media quality. Examples include scientific and medical applications such as scans or microscope images and security surveillance video with fixed visual settings. Results from general semantic annotation of video and images can be found in the TRECVID[8] (Smeaton et al. 2006) and ImageCLEF[9] (Müller et al., 2010) evaluation workshops and for music information retrieval in MIREX[10] (Downie, 2008).

Automated annotation from features faces criticism owing to its current inability to model a large and useful vocabulary with high accuracy. Enser and Sandom (2003) argue that some of the vital information for significance and content of images has to come from metadata: it is virtually

[8] http://trecvid.nist.gov

[9] http://ir.shef.ac.uk/imageclef

[10] http://www.music-ir.org/mirex

impossible to, e.g., compute the date or location of an image from its pixels. A real-world image query such as “Stirling Moss winning Kentish 100 Trophy at Brands Hatch, 30 August 1968” cannot be answered without metadata. They argue that pixel-based algorithms will never be able to compute significance of images such as “first public engagement of Prince Charles as a boy” or “the first ordination of a woman as bishop”. Their UK-funded arts and humanities research project “Bridging the Semantic Gap in Visual Information Retrieval” (Hare et al. 2006; Enser and Sandom, 2003) brought a new understanding about the role of the semantic gap in visual image retrieval.

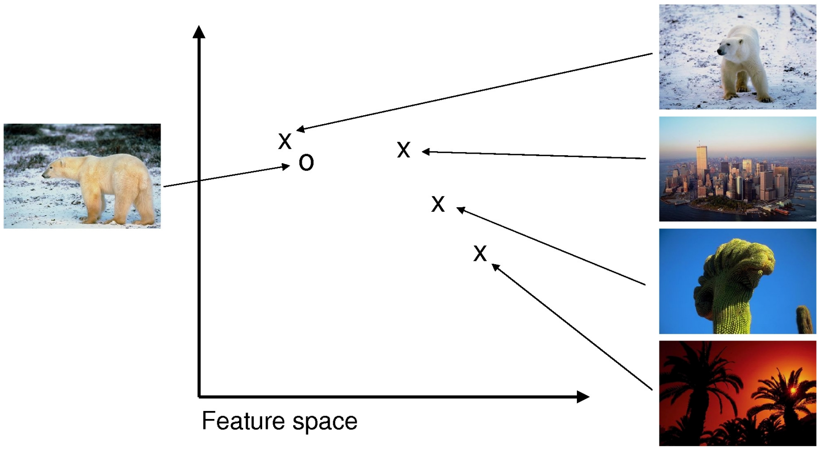
Content-based Media Retrieval

Content-based retrieval uses characteristics of the multimedia objects themselves, i.e., their content, to search and find multimedia. Its main application is to find multimedia by examples, i.e., when the query consists not of words but of a similar example instance. This technology differs radically from the library card paradigm as it tries to remove or reduce the need for formal text metadata.

One of the problems with finding media objects based only on the similarity of the low-level features is that *visual* (or audio) similarity doesn't necessarily imply *semantic* similarity. In spite of this there are a number of scenarios where query-by-example is very effective. IBM's QBIC project[11] was one of the earliest applications that enabled the querying of images by visual concepts such as colours, textures and positions without the need to describe them in words. The application of QBIC technologies within the Hermitage Museum site[12] allows browsers to find artworks based on their colour compositions. Empora.com[13] lets shoppers find fashion items such as clothes, shoes and accessories based on their colour or shape similarity. Shazam[14] will match specific music performances to identify a tune from a short recording. Numerous other applications, both commercial and research, have been implemented.

The basic principle behind query-by-example systems is to automatically extract descriptions of features such as colour, shape, texture, tempo, pitch etc. and when a query is submitted extract the same features from the query item. A numerical distance measure is then calculated to describe the similarity between the query object and the items in the media collection and the most “similar” objects are returns to the user. This is shown in figure 3 where the query is the image of a polar bear on the left. This query image will have a representation as a certain point (o) in feature space. In the same way, every single image in the database has its own representation (x) in the same space. The choice of features to build this feature space and how to compute distances has a vital impact on how well search by example works.

[11] http://wwwqbic.almaden.ibm.com/
[12] http://www.hermitagemuseum.org/fcgi-bin/db2www/qbicSearch.mac/qbic?selLang=English
[13] http://www.empora.com
[14] http://www.shazam.com/



3: Features and distances

The important bit is that both the query example and the database multimedia objects have undergone the same feature extraction mechanism. Figure 4 shows a typical architecture of such a system. The database indexing and similarity ranking tasks in Figure 4 identify the nearest neighbours in feature space with respect to a chosen distance function. It is sufficient to sort the distances of the query object to all database objects and only present as results the nearest neighbours, i.e., the ones with the smallest distances to the objects in feature space.

The ability to efficiently find nearest neighbours is an important part of implementing content-based retrieval systems. A more in-depth discussion of the issues facing multimedia indexing can be found in Rüger (2010), chapter 3. For efficiency reasons, approximate nearest neighbour indexing is often more useful with large datasets as it relaxes the constraint of finding an exact match in favour of speeding up the response. Muja and Lowe (2009) give a recent approach and provide example public domain code that implements it.

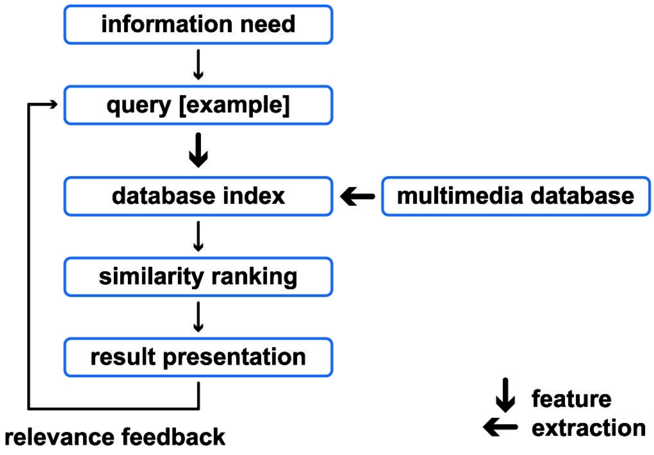
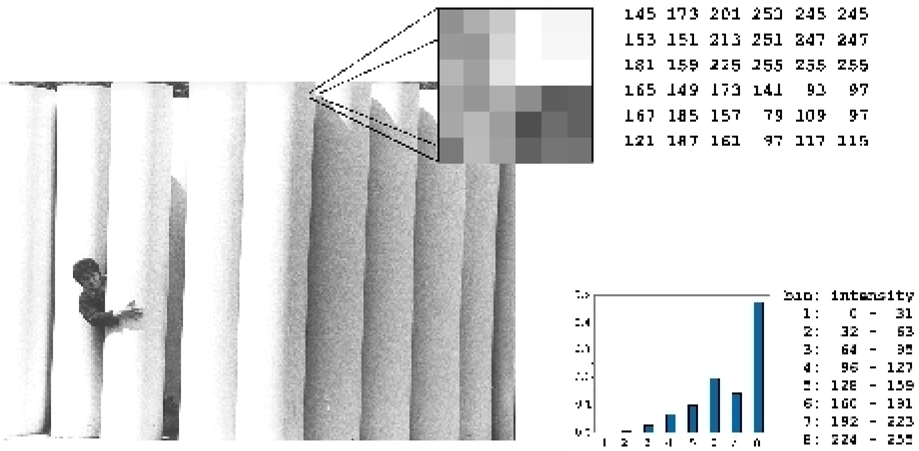


Figure 4: Content-based multimedia retrieval: generic architecture

Constructing an effective content-based retrieval system requires making a number of tradeoffs to ensure accuracy, robustness, responsiveness and scalability. Design choices to support the tradeoffs can be found in the selection of the best feature set and the distance metric used to calculate the similarity between the query object and the media collection in feature space. In the remainder of this section we will discuss some of the options relating to features and distance metrics for content-based media retrieval.



145	173	201	253	245	245
153	151	213	251	247	247
181	159	225	255	255	255
165	149	173	141	93	97
167	185	157	79	109	97
121	187	161	97	117	115

bin:	intensity
1:	0 - 31
2:	32 - 63
3:	64 - 95
4:	96 - 127
5:	128 - 159
6:	160 - 191
7:	192 - 223
8:	224 - 255

Figure 5: Millions of pixels with intensity values and the corresponding intensity histogram

Features

Automatically extracting suitable feature descriptors from media objects is a challenging task due to the sheer amount of data in multimedia documents often with apparently little meaningful structure. Look at the black and white photograph of Figure 5, for example. It consists of millions of pixels, and each of the pixels encodes an intensity (one number between 0=black and 255=white). One of the prime tasks in multimedia retrieval is to make sense out of this sea of numbers. The key here is to condense the sheer amount of numbers into meaningful pieces of information, which we call features.

One common way of indexing multimedia is by creating summary statistics, which represent colour usage, texture composition, shape and structure, localisation, motion and audio properties. In this section, we discuss some of the widely used features and methods in more depth. Deselaers et al (2008) compare a large number of different image features for content-based image retrieval and give an overview of a large variety of image features. Features are not only relevant for retrieval tasks: Little and Rüger (2009) demonstrate how important the choice of the right features is for the automated image annotation task.

Colour Histograms

Colour is a phenomenon of human perception. From a purely physical point of view, light emitted from surfaces follows a distribution of frequencies to create, for example, ‘yellow’ butter, ‘red’ tomatoes and ‘green’ lettuce. Each pure spectral frequency corresponds to a hue, all of which create the rainbow spectrum. The human eye has three different colour receptors that react to three different overlapping ranges of frequencies. Their sensitivity peaks fall into the red, green and blue areas of the rainbow spectrum. Hence, human perception of colour is three-dimensional, and modelling colour as a mixture of red (R), green (G) and blue (B) is common. Virtually all colour spaces are three-dimensional (except for ones that utilise a fourth component for black), and therefore so are colour histograms used to represent the colour description.

An example of a 3-dimensional colour histogram is depicted in Figure 6, which shows a crude summary of the colour usage in the original image. Here each of the red, green and blue colour axes in the so-called RGB space is subdivided into intervals yielding 4 × 4 × 4 = 64 3d colour bins; the proportion of pixels that are in each bin is represented by the size of a circle, which is positioned at the centre of a bin and coloured in correspondingly.

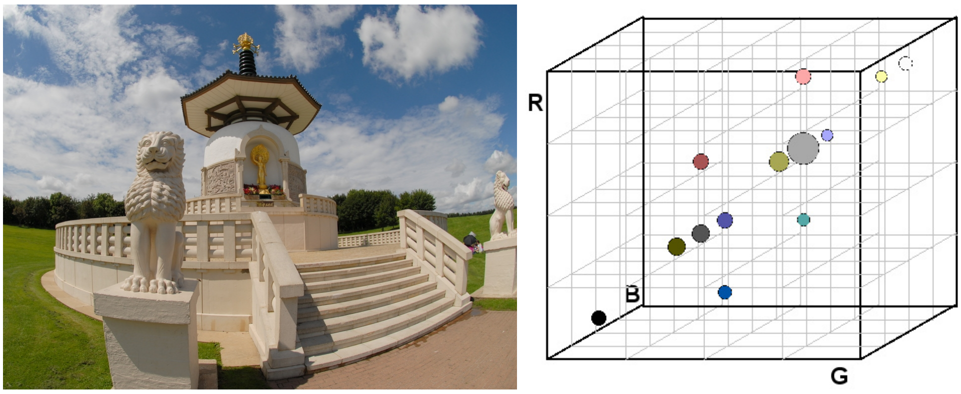


Figure 6: 3D colour histogram

The 3d colour histogram is represented as a vector of real numbers calculated by partitioning the image into cells and counting the number of occurrences in each cell. The histogram vector can be wrapped in XML format for MPEG-7 as shown by the fragment in figure 7 - a normalised colour descriptor example generated from the PHAROS project (Bozzon et al. 2009). Details about general algorithms for calculating colour histograms can be found in Rüger (2010).

```
<MediaUri>file://server/location/380942_1.jpg</MediaUri>
[...snip...]
  <AttributeValuePair>
    <Attribute href="urn:x-mpeg-7-pharos::image:kmi:margCIELAB-2+2+2-M-G.T3x3"/>
    <FloatMatrixValue dim="54">0.9328 0.0783 -0.0148 0.0154 0.0211 0.0743 0.9439
0.0188 -0.0134 0.0117 0.0180 0.0478 0.9319 0.0807 -0.0126 0.0124 0.0141 0.0532 0.8983
[...snip...]
0.0087 -0.0127 0.0083 0.0120 0.0304 0.9306 0.0804 -0.0140 0.0145 0.0179 0.0691 0.0691
    </FloatMatrixValue>
  </AttributeValuePair>
```

Figure: 7 MPEG-7 example for feature-based annotation of an image

RGB is not the only representation of colour space. A common alternative is hue (H) and saturation (S) based models that use variation (HSV), luminance (HSL) or brightness (HSB). These give a continuous representation of colour so that colours at the ends of the spectrum (e.g., red) can have two numerically distinct values (e.g., red = (1,100,50) = (360,100,50)). While this creates difficulties when using absolute differences to calculate similarity it does allow more natural differences such as “brighter” or “purer” colours to be defined.

The CIE (International Commission on Illumination/Commission Internationale de l'éclairage) series of colour models, which aim to more closely approximate human perception of colour, are also commonly used in automatic annotation and content-based indexing – e.g., CIELAB/CIELUV. Luminance plus chrominance models (e.g., YUV, YPbPr, YCbCr) are less common as they are mostly used for digital broadcast applications or compression but also possible representations of colour. Models specifically developed for printing colour (e.g., CMYK) are rarely used. Sangwine and Horne (1998) and Busin et al. (2008) provide some overviews explaining and comparing colour space models.

Colour descriptors included in the MPEG-7 standard are color space, dominant color, scalable color, group of frames, color structure and color layout (See Messing et al. 2001).

Texture Histograms

Simple colour descriptors are rarely sufficient to index a multimedia collection and other, more complex, feature descriptions such as texture are often used. Tamura et al. (1978) found out through psychological studies that humans respond best to coarseness, contrast, and directionality and to a lesser degree to line-likeness, regularity and roughness. Unlike colour, which is a property of a single pixel, texture is a property of a region of pixels, so we need to look at an area around a pixel before we can assign a texture to that pixel. By considering a window around each point in an image, values for coarseness (C), contrast (N) and directionality (D) can be calculated. Other texture descriptors include Gabor filters (Turner, 1986), co-occurrence matrix (or Haralick features (Haralick et al., 1973)) and wavelet transforms. Manjunathi (1996), Grigorescu et al. (2002) and Howarth and Rüger (2005b) compare the performance of selected texture descriptors.

Texture descriptors included in the MPEG-7 standard are homogeneous texture, texture browsing (analogous to the discussed human perceptual characterization, such as regularity, coarseness, directionality) and edge histogram.

Shape

Shape is commonly defined as an equivalence class of geometric objects invariant under translations, rotations and scale changes that keep the aspect ratio. Many retrieval applications require global scale invariance, i.e., relative sizes are still meaningful. Strict scale invariance would imply that the sizes of objects do not matter at all. For example, the size of a face in close up would be very different to the size of the same face contained within a line up. We will describe some examples of scale invariant features as used to determine points of interest in the later section on global and local features.

Shape representations can preserve the underlying information, so the shape can be reconstructed (e.g., for compression) or they can aim to describe interesting features for analysis, indexing or retrieval. There are a number of boundary-based features that can be extracted — these ignore the interior landscape of segments including holes in it. Other shape features are region-based.

The boundary of a 2d shape can be calculated and described in a number of ways. The most straightforward is to calculate the perimeter of the shape, however if you are counting pixels then this is insufficient as the digitisation of a mathematical line consumes different amounts of pixels at different angles and apparent line thicknesses. More commonly, measures that describe the relative shape of the boundary such as convexity, circularity or the number of corners calculated and described using Freeman chain codes, difference chain codes or Fourier descriptors are used. More information about calculating boundary descriptors can be found in Rüger (2010).

By creating a mathematical model of the pixels that make up a shape it is also possible to calculate features that describe the region as a whole rather than only its contour or boundary. This enables descriptions that distinguish between, for example, circular and doughnut-shaped objects among others. Examples include: eccentricity, orientation, area, centre of mass (or centroid) and minimal bounding box that are described with real numbers or real number vectors. Many of these descriptors can be extracted using tools such as MATLAB and are defined and explained in its documentation[15]. More elaborate shape representations have also been developed some of which are the curvature scale space representation or the spline curve approximation that require sophisticated shape matching techniques (del Bimbo and Pala, 1997).

Shape descriptors specifically included in the MPEG-7 standard are contour-based (what we have termed boundary-based), region-based, 3-D and multiple-view.

[15] http://www.mathworks.com/help/toolbox/images/ref/regionprops.html

Video features

In most cases videos are analysed using the same visual features as still images – i.e., colour, shape, texture. Generally, a subset of frames (images) is extracted from the video to be analysed and indexed. This set of keyframes can be chosen by dividing a video into shots and selecting a representative frame for each shot. The shot boundary detection problem has attracted much attention as it is an essential pre-processing step to virtually all video analysis. Smeaton et al. (2009) give an overview of the TRECVid shot boundary detection task summarising the most significant of the approaches taken over the years from 2001 to 2007.

A shot has a technical definition as the smallest unit of continuous filming before a cut or other more gradual transition occurs. Abrupt cuts, in particular, can be identified automatically by analysing the changing features of the surrounding frames. These are then useful for generating summaries, browsing or generating a set of keyframes for further analysis and indexing. Shots themselves don't always correspond to a semantic unit – this is formed by a scene, which is often a collection of shots. Different types of video can be composed of very different numbers of shots. Consider how many distinct shots exist in a high-energy music video as compared with a movie such as Russian Ark (filmed as a single, long shot). The number or rate of shots can itself be considered a feature description.

MPEG-7 also includes motion descriptors specifically for video: motion activity, camera motion, motion trajectory and parametric motion (including translational parameters, rotational/scaling parameters, and perspective parameters). Shots and scenes are defined as video segments, a type of temporal decomposition, within a larger video document. Keyframe images are attributes of the video segments.

Audio features

While we have focused on visual features there are also features for audio (music, soundtracks, speech etc.) to enable tools to search and filter audio content in regard to, e.g., spectrum, harmony, timbre and melody. The results can be used in much the same way as the visual feature descriptors that have been described in more detail. A good range of audio features are reviewed by Liu (1998) and Foote (1999). MPEG-7 also contains a number of specific low-level audio descriptors.

Describing Feature Histograms

Describing a set of values using a histogram rather than a vector reduces the number of values required. This results in a smaller set of numbers to describe a media object or reduces the dimensionality of the descriptors. Lower dimensionality speeds up the calculation of distance between features sets when searching for the nearest neighbours and provides a more usable abstraction. However it is also possible to further summarise the histogram vector. Both colour and texture histograms describe the distribution of particular characteristics across the pixel matrix that represents an image. Statistical moments are an alternative way to summarise distributions.

If you wanted to express a quality of an object, say the intensities of image pixels in a particular area, through one number alone you would most likely choose its average, which produces a single value describing essentially the dominant colour intensity for that region. Other moments include the variance, skewness and kurtosis of a distribution. Altogether this small set of numbers can represent the shape of the histogram and replace a lengthy (possibly sparse) vector with four values.

Global and Local Features

So far we have only considered features that are calculated across the entirety of an image – *global* features. While these are useful they are very crude indicators of similarity. For example an eight-bin histogram of intensity values of an image is only a simple approximation of its brightness distribution, and many images share the same histogram. Figure 3 shows a woman in the middle of bright column sculptures, but an image of a skier in snow is likely to have the same intensity histogram. The other disadvantage of global histograms computed over the whole image is that they lose all locality information – e.g., that the top corner is bright and the lower half is dark.

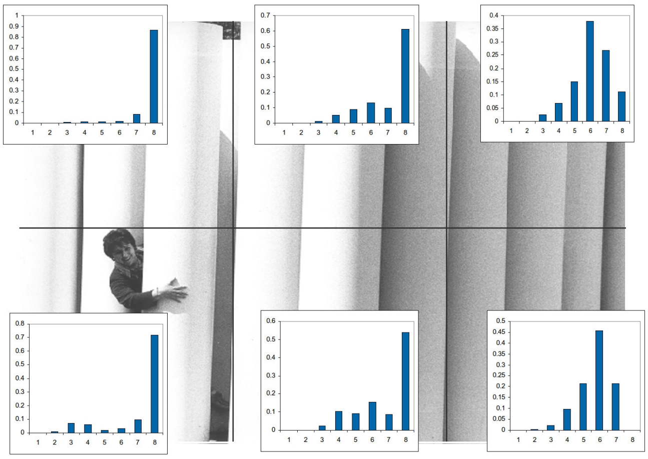


Figure 8: The intensity histograms of a tiled image

One simple solution is to tile an image into $n \times m$ rectangles of the same size, each of which creates a histogram (see figure 8). The full feature vector is then the concatenation of the feature vectors of individual tiles and, in this case, contains $6 \cdot 8$ numbers. It is also possible to divide images up into regions of different size to weight the importance of certain areas. For example, a centre-border intensity histogram, where two histograms are computed: the centre area much bigger than the border area. The corresponding proportion of pixels that fall into centre area intensity ranges is typically larger than for intensity ranges of the border area. This gives the centre extra weight. Other divisions of the image can be considered that adhere to general photographic principles such as placing objects of interest in accordance with the “rule of thirds” or “golden ratio”. These principles are well known in photography and divide an image mathematically to stipulate the optimum position for objects of interest. Some example tiling, with points marked according to the rule of thirds and golden ratio, are shown in figure 9.

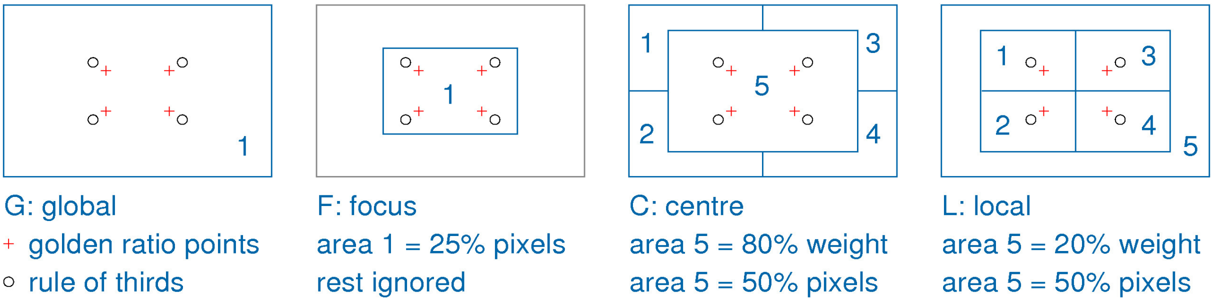


Figure 9: Different strategies to capture essential areas in photographs

Incorporating spatial awareness into global histogram calculation creates finer discrimination between images and maintains some local information. Another approach is to exploit local structure to identify regions or points of interest. These are salient points in images that give rise to encoding features in limited area. Figure 10 illustrates the idea, but algorithms that compute points of interests normally compute many more regions.

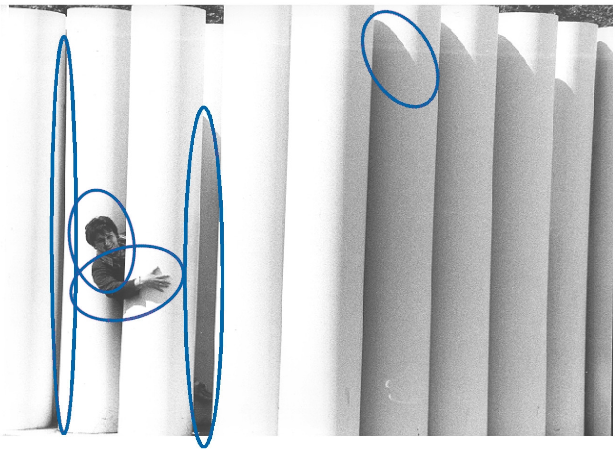


Figure 10: Points and regions of interest

There are a number of approaches for identifying points of interest including using object segmentation, edge detection and mapping the rate of change of intensity, colour or texture. However when indexing collections of media what you really want is to be able to identify similarities independently of their scale (e.g., size) within the image. Feature descriptors that are invariant to changes in scale or perspective and robust to small variations in lighting or colour are very popular in content-based multimedia retrieval.

Lowe (1999, 2004) developed a popular scale-invariant feature transform (SIFT) that detects and encodes local features in images. The first step is to detect candidate points of interest in an image by convolving a 2d Gaussian function of different scale σ with the image (basically a smoothing or blurring of the image with different radius) and taking differences of the resulting function (blurred image) at each point with respect to slightly different scales σ and $k\sigma$. The extrema of this function, called difference of Gaussians, indicate candidate points of interest. From these points *keypoints* are localised and orientations assigned, which serve as a local reference coordinate system. The final features that are extracted from this area are computed relative to this local reference, so that they are encoded in a scale, rotation and location invariant manner. A typical image exhibits in the order of 2000 key-points. A key-point is described with a vector of 128 real numbers.

SIFT features require significant computational effort to calculate from an image and are described using a lot of numbers which makes them slow to compare. To overcome this Bay et al. (2006) proposed speeded up robust features (SURF) that use integral images (similar to summed area tables) to quickly compute box-type convolution filters. Points of interest are detected based upon an approximated Hessian matrix by using box-filters to approximate the second-order Gaussian derivatives. Using integral images these can be calculated at low cost independent of filter size. SURF key-point descriptors consist of 64 dimensions – quicker than SIFT to calculate and compare. Figure 10 shows SIFT and SURF keypoints displayed on the same image. You can see how each method has identified roughly similar general areas of interest. Both methods can be configured to alter their sensitivity. In this instance more SURF points appear to have been detected but as SURF descriptors (64) are smaller than SIFT descriptors (128), they can still be more efficient to index.

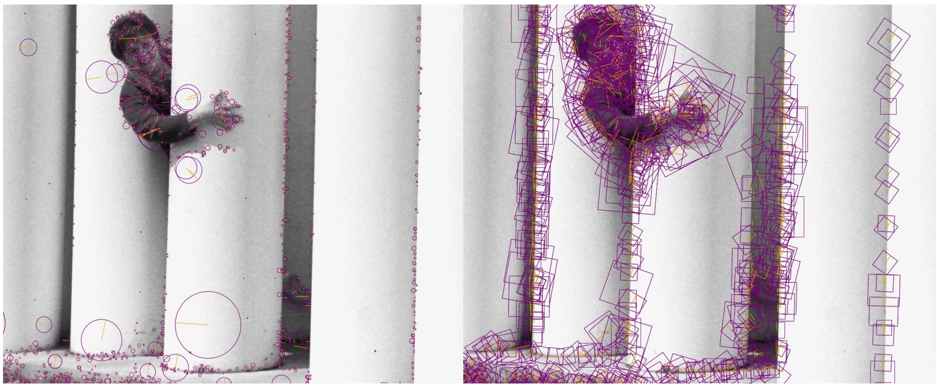


Figure 11: SIFT (L) and SURF (R) keypoints

Different methods for calculating invariant features including SIFT/SURF based on the colour channels rather than only intensity values are discussed in van de Sande et al. (2010).

Using points of interests with local features allows one to quantise the features into so called visual words. Being quantised one can deploy the same type of retrieval techniques that are so successful for text retrieval: inverted-file indices. This gives faster, more responsive and scalable search implementations. Localised features and quantisation is a very powerful combination for duplicate detection and fingerprinting.

Fingerprinting, or known-item search, is a specific application of content-based multimedia retrieval. In certain searches there is the desire to match not the general type of scene or music that is represented in the query object (e.g., a sunset, pop or classical music) but instead to find variations or versions of that exact object (e.g., my holiday photo, copies of the recording of The Beatles, Come Together from the Abbey Road album). Multimedia fingerprints are descriptors (sets of features) that enable the unique identification of multimedia objects in a database, given a possibly different representation of it – e.g., a cropped image, re-formatted video or re-used audio fragment. Fingerprints should be small and allow the fast, reliable and unique location of the database record – they are not intended to be understood by people but used only as an indexing tool. Most importantly they are robust against degradation or deliberate changes as long as the human perception is the same. Fingerprinting is often used in copy-detection tasks

Since we want to identify objects based on human perception, simple hashing approaches that summarise an exact digital object, such as MD5 etc., are not sufficient. Datar et al.(2004), Cano et al. (2005) and Rüger (2010) discuss specific approaches to audio and visual fingerprinting.

Querying and Distance

The multimedia feature descriptors discussed so far can be used to index a collection of media for use in a content-based retrieval system. Once this index has been created it can be searched by taking an example media object (e.g., a query image) and extracting the same feature descriptors from it. A method is then needed for ranking the indexed media according to their similarity to the query object – their distance from each other in feature space. Feature space is the multi-dimensional area mathematically defined by the feature descriptors. Most of above features create real-valued number vectors of a fixed dimension, where distance (or difference) computation is straightforward.

The most common methods calculate geometric component-wise distances based on the Minkowski norm – Manhattan (or taxi cab), Euclidean, Chebyshev (or max norm). These measures place varying importance on the degree of difference between the values. Max norm distance is rarely useful as it ensures that the features with the greatest difference dominate the calculation of the distance value so that a pair of very similar media objects with one highly different feature will have a very large distance value. Manhattan (the number of “blocks” between two points in space) or Euclidean (the length of a straight line between two points in space) distance measures are more common. Hu et al. (2008) and Makadia (2008) evaluate the effectiveness of different distance metrics while Howarth and Rüger (2005a) discuss the usefulness of variations on standard Minkowski distances.

Other distance metrics are used for specific types of features including strings, ordinals/nominals or probabilistic measures. A more detailed discussion of distance metrics, including the necessity for normalising or standardising feature descriptors can be found in Rüger (2010).

Querying in this way results in an ordered list of indexed media objects each with a numerical distance value describing its “similarity” to the query image. The creation of this ranked list forms the core of a content-based multimedia retrieval system. In this section we have briefly summarised some of the main types of feature descriptors for multimedia and a few of the common methods for calculating similarity using mathematical distance measurements. These provide the basic building block for a multimedia search system.

Building a multimedia search system

The first step when dealing with the representation and retrieval of multimedia information in relation to image, audio and video search is to decide what type of queries need to be supported and what point along the multimedia metadata scale (shown in figure 1) will best enable these queries. In straightforward cases where the multimedia objects are to be treated as a catalogue and simple bibliographic data is sufficient then building a search system doesn't differ greatly from any other traditional text indexing system or digital library application.

Generally, multimedia search systems are interested in fully exploiting the rich information contained in media objects and higher levels of metadata or access to content-based queries will be desired. Design decisions will need to be taken to optimise the system's performance and increase the types of queries that it can address. If text-based queries are desired, either through piggy-backing, manual or automatic semantic annotation, then the scope, accuracy and expense of creating a text-based index for the multimedia collection need to be judged depending on the purpose. For example, in an online shopping catalogue: is the creation date for the media important or will the date of entry into the system suffice.

If the types of interaction are best served through content-based multimedia indexing and retrieval techniques then some of the decisions that need to be made include:

1. The best features for the type of media and the queries weighing the competing needs for processing speed (to extract the features and calculate the neighbours), flexibility and precision.
2. The most efficient method for summarising (reducing the dimensionality), normalising and storing the feature vectors.

3. The best method for measuring similarity between media objects using the set of features – the distance metrics that can be calculated.
4. Handling the trade-offs between responsiveness, scalability and query expressivity.
5. Using an efficient implementation of the nearest neighbour algorithm to search the indexed media collection.

In this chapter we have given a brief overview of methods for multimedia indexing and search. The focus has been on the use of content-based retrieval, which differs from traditional text-based indexing and querying, allowing media objects to be used as queries to multimedia collections. With ever increasing quantities of media objects being produced and used in a wide variety of applications – commercial, scientific, cultural, entertainment, personal etc. – representing and indexing large media collections is a common challenge.

References

Bay, H.; Tuytelaars, T. & Van Gool, L. (2006), Surf: Speeded up robust features, *in* 'Proceeding of the European Conference on Computer Vision', Springer, pp. 404-417.

del Bimbo, A. & Pala, P. (1997), 'Visual Image retrieval by elastic matching of user sketches', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**(2), 121-132.

Boll, S.; Klas, W. & Sheth, A. (1998), 'Overview on using metadata to manage multimedia data', Multimedia Data Management: Using Metadata to Integrate and Apply Digital Media. McGraw-Hill Publishers, 1-23

Bozzon, A.; Brambilla, M.; Fraternali, P.; Nucci, F.; Debald, S.; Moore, E.; Neidl, W.; Plu, M.; Aichroth, P.; Pihlajamaa, O.; Laurier, C.; Zagorac, S.; Backfried, G.; Weinland, D. & Croce, V. (2009), PHAROS: an audiovisual search platform, *in* 'ACM International Conference on Research and Development in Information Retrieval', pp. 841.

Busin, L.; Vandenbroucke, N. & Macaire, L. (2008), 'Color spaces and image segmentation', *Advances in imaging and electron physics* **151**, 65--168.

Cano, P.; Batlle, E.; Kalker, T. & Haitsma, J. (2005), 'A review of audio fingerprinting', *The Journal of VLSI Signal Processing*, Springer, **41**, 271-284

Datar, M.; Immorlica, N.; Indyk, P. & Mirrokni, V. (2004), Locality-sensitive hashing scheme based on p-stable distributions, *in* 'ACM Annual Symposium on Computational Geometry', pp. 253--262.

Datta, R.; Joshi, D.; Li, J. & Wang, J. (2008), 'Image retrieval: ideas, influences, and trends of the new age', *ACM Computing Surveys* **40**(2), 1--60.

Deselaers, T.; Keysers, D. & Ney, H. (2008), 'Features for image retrieval: an experimental comparison', *Information Retrieval* **11**(2), 77--107.

Doraisamy, S. (2005), 'Polyphonic music retrieval: the n-gram approach', PhD thesis, Imperial College London.

Downie, S. (2008), 'The music information retrieval evaluation exchange (2005-2007): a window into music information retrieval research.', *Acoustical Science and Technology* **29**(4), 247--255.

Downie, S. & Nelson, M. (2000), Evaluation of a simple and effective music information retrieval method, *in* 'ACM International Conference on Research and Development in Information Retrieval', pp. 73--80.

Enser, P. & Sandom, C. (2003), Towards a Comprehensive Survey of the Semantic Gap in Visual Image Retrieval, *in* 'International Conference on Image and Video Retrieval', Springer LNCS 2728, pp. 163--168.

Finke, R. (1989), *Principles of Mental Imagery*, Cambridge MIT Press.

Foote, J. (1999), 'An Overview of Audio Information Retrieval', *Multimedia Syst* **7**(1), 2--10.

Grigorescu, S.; Petkov, N. & Kruizinga, P. (2002), 'Comparison of texture features based on Gabor filters', *IEEE Transactions on Image processing* **11**(10), 1160--1167.

Haralick, R.; Shanmugam, K. & Dinstein, I. (1973), 'Textural features for image classification', *IEEE Transactions on systems, man and cybernetics* **3**(6), 610--621.

Hare, J.; Lewis, P.; Enser, P. & Sandom, C. (2006), Mind the gap: another look at the problem of the semantic gap in image retrieval, *in* 'Multimedia Content Analysis, Management and Retrieval, SPIE Vol 6073', pp. 1--12.

Howarth, P. & Rüger, S. (2005b), 'Robust texture features for still-image retrieval', *IEE Proceedings on Vision, Image and Signal Processing* **152**(6), 868--874.

Howarth, P. & Rüger, S. (2005a), Fractional distance measures for content-based image retrieval, *in* 'European Conference on Information Retrieval', Springer LNCS 3408, pp. 447--456.

Hu, R.; Rüger, S.; Song, D.; Liu, H. & Huang, Z. (2008), Dissimilarity measures for content-based image retrieval, *in* 'IEEE International Conference on Multimedia and Expo', pp. 1365--1368.

Jung, K; Kim, K. I. & Jain, A. K. (2004), 'Text information extraction in images and video: a survey', Pattern Recognition, 37(5), 977-997

Lagoze, C. & Payette, S. (2000), Metadata: principles, practices and challenges, *in* A Kenney & O Rieger, ed., 'Moving Theory into Practice: Digital Imaging for Libraries and Archives', Research Libraries Group, Mountain View, CA.

Lew, M. S.; Sebe, N.; Djeraba, C. & Jain, R. (2006), 'Content-based multimedia information retrieval: State of the art and challenges', *ACM Trans. Multimedia Comput. Commun. Appl.* **2**(1), 1--19.

Little, S. & Rüger, S. (2009), Conservation of effort in feature selection for image annotation, *in* 'IEEE Workshop on Multimedia Signals Processing'.

Liu, Z.; Wang, Y. & Chen, T. (1998), 'Audio feature extraction and analysis for scene segmentation and classification', *VLSI Signal Processing* **20**(1--2), 61--79.

Lowe, D. (2004), 'Distinctive image features from scale-invariant keypoints', *International Journal of Computer Vision* **60**(2), 91--110.

Lowe, D. (1999), Object recognition from local scale-invariant features, *in* 'IEEE International Conference on Computer Vision', pp. 1150--1157.

Makadia, A.; Pavlovic, V. & Kumar, S. (2008), A new baseline for image annotation, *in* 'European Conference on Computer Vision', Springer LNCS 5304, pp. 316--329.

Manjunath, B. (1996), 'Texture features for browsing and retrieval of image data', *IEEE Transactions on pattern analysis and machine intelligence* **18**(8), 837.

Manjunath, B.; Salembier, P. & Sikora, T. (2002) *Introduction to MPEG-7: multimedia content description interface* John Wiley & Sons Inc

Messing, D.; van Beek, P. & Errico, J. (2001), The MPEG-7 colour structure descriptor: image description using colour and local spatial information, *in* 'International Conference on Image Processing', pp. 670--673.

Muja, M. & Lowe, D. (2009), Fast approximate nearest neighbors with automatic algorithm configuration, *in* 'International Conference on Computer Vision Theory and Applications', pp. 331--340.

Müller, H.; Clough, P.; Deselaers, T. & Caputo, B., ed. (2010), *ImageCLEF: Experimental Evaluation in Visual Information Retrieval*, Springer.

Nack, F. & Lindsay, A. (1999), 'Everything you wanted to know about MPEG-7. Part 2', *IEEE Multimedia*, **6**, 64-73

Rüger, S. (2010), *Multimedia information retrieval*, Morgan & Claypool Publishers.

van de Sande, K. E. A.; Gevers, T. & Snoek, C. G. M. (2010), 'Evaluating Color Descriptors for Object and Scene Recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(9), 1582--1596.

Sangwine, S. & Horne, R. (1998), *The colour image processing handbook*, CRC Press.

Schallauer, P.; Bailer, W. & Thallinger, G. (2006), 'A description infrastructure for audiovisual media processing systems based on MPEG-7', *Journal of Universal Knowledge Management* **1**(1), 26--35.

Smeaton, A.; Over, P. & Doherty, A. (2009), 'Video shot boundary detection: seven years of TRECVid activity', *Computer Vision and Image Understanding*.

Smeaton, A.; Over, P. & Kraaij, W. (2006), Evaluation campaigns and TRECVid, *in* 'ACM International Workshop on Multimedia Information Retrieval', pp. 321--330.

Smeulders, A.; Worring, M.; Santini, S.; Gupta, A. & Jain, R. (2000), 'Content-based image retrieval at the end of the early years', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(12), 1349--1380.

Tamura, H.; Mori, S. & Yamawaki, T. (1978), 'Textural features corresponding to visual perception', *IEEE Transactions on Systems, Man and Cybernetics* **8**(6), 460--472.

Troncy, R.; Celma, O.; Little, S.; Garcia, R. & Tsinaraki, C. (2007), MPEG-7 based Multimedia Ontologies: Interoperability Support or Interoperability Issue?, *in* 'Proceedings of Workshop on Multimedia Annotation and Retrieval Enabled by Shared Ontologies (SAMT)'.

Turner, M. (1986), 'Texture discrimination by Gabor functions', *Biological Cybernetics* **55**(2), 71--82.

Witten, I. H.; Bainbridge, D. & Nichols, D. M. (2009), *How to Build a Digital Library, Second Edition (The Morgan Kaufmann Series in Multimedia Information and Systems)*, Morgan Kaufmann.

Zeng, M. & Qin, J. (2008), *Metadata*, Neal-Schuman, New York.

Zhang, J. & Kasturi, R. (2008), 'Extraction of text objects in video documents: Recent progress', The Eighth IAPR International Workshop on Document Analysis Systems, 5-17

Zhao, R. & Grosky, W. (2001), 'Bridging the semantic gap in image retrieval', *Distributed multimedia databases: Techniques and applications*, Idea Group Publishing, 14-36