State of the Art in content description and access methods D 2.0

Project Number:	FP6 - 045389
Deliverable id:	D 2.0
Deliverable name:	Report on mono-media up-to-date state-of-the-art (cross different sub-tasks)
Date:	25 September 2007

Information Society Technologies

COVER AND CONTROL PAGE OF DOCUMENT

Project Acronym:	VITALAS
Project Full Name:	Video & image Indexing and Retrieval in the Large Scale
Document id:	D 2.0
Document name:	Report on mono-media up-to-date state-of-the-art (cross different sub-tasks)
Document type (PU, INT, RE)	RE
Version:	V 0.6
Date:	25 September 2007
Authors: Organisation: Email Address:	A. Joly INRIA alexis.joly@inria.fr

Document type PU = public, INT = internal, RE = restricted

ABSTRACT:

This deliverable overviews the state of the art in mono-media content indexing. A wide range of literature has been reviewed in the course of the associated workpackage WP2. Media for which previous work is analysed in depth are still-image, video, audio, and text as well as efficient and similarity search structures. The aim of this deliverable is to guide the specification of the modules of VITALAS prototype V1 and to justify further researches.

KEYWORD LIST: mono-media indexing, Image and Video content description, Audio content description, text description, low level features, indexing structures, similarity search structures

List of Contributors

- Alexis Joly, INRIA
- Olivier Buisson, INA
- Bruno GRILHERES, EADS
- Gérard DUPONT, EADS
- Alberto Isasi ANDRIEU, ROBOTIKER
- Michael Oakes, UoS
- Mustafa AbuSalah, UoS
- Marco Palomino, UoS
- Yan Xu, UoS
- Christos Papachristou, CERTH-ITI
- Panagiotis Panagiotopoulos, CERTH-ITI
- Christos Diou, CERTH-ITI
- Daniel Schneider, Fraunhofer IAIS
- Arjen de Vries, CWI

Table of contents

1 Introduction

Information Retrieval systems are conventionally divided into two main components: an indexing engine which works continuously in the background to extract what we will call here document signatures and stores them in an index database; and an interactive query or retrieval engine which allows searchers to type in queries, browse search results and select queries, present example “relevant” documents and so on.

The offline indexing processes allows the retrieval engine to operate at interactive speeds. Without it would not be possible to provide an acceptably rapid interactive experience for the searcher with the scale of document numbers needed by systems like VITALAS. It relies on the signatures being good representations of the documents for the purpose of retrieval: in other words they should (for most queries and most searchers, most of the time) allow relevant and irrelevant documents to be distinguished easily and with a minimal amount of stored data.

For VITALAS many documents are composites of several different media (natural language annotations, keywords annotations, audio, images and video). The first major objective of the VITALAS project is "Cross-media indexing and retrieval", as outlined in Annex I - "Description of Work". The first stage in achieving this objective is the process of extracting features from the raw incoming media and possibly the generation of description and annotation data for each medium separately. In VITALAS this is the one of the focus of WP2, and this literature review therefore focuses on the state of the art in the production of mono-media features. Each media is developed in a separate section of this document: Section 2 focuses on visual content description techniques including video and still images. Section 3 deals with audio content description techniques and Section 4 is concerned with text indexing techniques.

The second major objective of the VITALAS project is to enable the use of very large documents databases, up to several millions of still images and several ten thousand hours of video materials. This scalability issue requires efficient access methods to the produced mono-media features, and this literature review therefore also focuses on the up-to-date state of the art in similarity search techniques and indexing structures. This topic will be developed in Section 5.

The aim of this deliverable is to guide the specification of VITALAS V1 modules and to focus the further research works of the project. It is not an exhaustive survey of the mentioned topics since this has already been done in many other literatures. We pay more attention to up-to-date progresses beyond previous state-of-the-art reviews and we discuss the usefulness of older techniques regarding modern works and applications that are in the scope of VITALAS.

2 Visual content

The visual content of multimedia documents can be described either by low level features that represent some signal level statistics of the image content, such as texture, shapes, etc. or by higher level visual concepts such as recognized objects or identified faces. Notice that the higher level concepts are generally obtained from previously extracted low level features and not directly from the raw document. In the following, we first review and comment most used low level visual features (2.1) and we then focus on higher level visual concepts with a focus on object detection and recognition techniques (2.2) and face detection and recognition techniques (2.3).

2.1 Low level visual features

2.1.1 Global features

The quality of the low-level descriptors used in any CBIR or automatic image annotation system is crucial since the extracted visual signatures are the raw material on which all the visual algorithms of our domain are based to express visual similarity or to find some semantic concept. Designing visual descriptors for specific databases, with a priori knowledge that can be encapsulated in the descriptor's definition is a difficult task but finding good descriptors definition for generic databases is even more challenging. Colours, textures and shapes have been identified as the main low-level aspects that can be characterized in images. Among the earliest use of colour histograms for image indexing was that in [Swain1991]. Subsequently, feature extraction in systems such as QBIC [Flickner1995], Pictoseek [Gevers2000], and VisualSEEK [Smith1997] are notable. Innovations in colour constancy were made by including specular reflection and shape into consideration [Finlayson1996]. In [Huang1999] colour correlograms were proposed as enhancements to histograms that take into consideration spatial distribution of colours as well. Daubechie's wavelet transforms were used for texture feature extraction in the WBIIS system [Wang1998]. The visual features included in the MPEG7 standard are among the most famous. They consist in a set of colour and texture well suited to natural images and video [Manjunath2001]. These include histogram-based descriptors, spatial colour descriptors and texture descriptors. More details can be found in [salembier2002].

Global descriptors have been used for a long time to characterize the visual aspect of images. Nowadays, the design of global features is less studied as people are focussing on local feature approaches [Datta2007]. However, global features are still used in a lot of recent image retrieval works and automatic image annotations schemes [Yavlinsky2005], [Herve2007], [Li2006], [Grira2006], [Murphy2006], [Liu2007]. This is due to the fact that global features still encounter some crucial advantages:

- They are faithful to the image content while requiring a low amount of data to describe an image.
- The processing time required for extracting and comparing global features is low.
- Most global features being histograms, they can be easily combined and used simultaneously, with the same distance functions or the same kernels.
- They are as effective as or even more effective than local features for certain applications involving high level global semantics (e.g. "outdoor scene", "urban scene", "night", "painting", etc.) or subjective concerns (trends, harmony, ambiance, etc.).

Even in very recent image retrieval benchmarks, global features still show very good performances. The global descriptors developed by VITALAS partner INRIA, for example, ranked first in the image annotation task of the 2006 ImagEVAL campaign [Moellic2006]. ImagEVAL is an initiative that was launched in France in 2006 and whose interesting aspect is that its specification and organization were established by both a research team and professional archivists. We are therefore close to real-life scenarios, with challenging image collections. Several European teams participated in the evaluation, as well as private companies. Most of the descriptors used for this benchmark already existed in the visual search engine of the team IKONA [Boujemaa2001]. They have already been widely tested in visual similarity search and relevance feedback scenarios. New results on scene categorization and object recognition on some common publicly available research databases were also obtained in the scope of VITALAS WP2 experiments [Herve2007]. They show that a simple SVM approach coupled with the global features of IKONA perform very closely than best recent object class recognition techniques for some well known evaluation databases (Corel2000, Caltech4, Xerox7, PASCAL VOC2005, ImagEVAL task4). Only most recent challenging databases such as Caltech101 and PASCAL VOC2006 lead to more degraded results comparing to local features based approaches.

As IKONA descriptors will probably be integrated as a state-of-the-art technique in VITALAS prototype, we give here some details of these features. For the colour description, we use a standard HSV histogram and weighted colour histograms [Vertan2000] which make it possible to combine both colour and structure information in a single representation. It is well known that usual colour histograms do not keep any spatial information about the pixels. But it is also known that pixels with the same colour do not have an equal visual importance depending on their localization in the image. Thus it is useful to include some local activity information, measuring local uniformity or non-uniformity, in colour histograms. Shape and colour are merged by weighting pixel colours with the Laplacian. Texture and colour are merged by weighting colours with a probability measure. Texture information is gathered by a Fourier histogram [Ferecatu2005]. After obtaining the 2D Fourier transform of an image, two histograms are computed in the complex plane. They represent two types of distributions of the energy. The first histogram is computed with a circular partition, the second uses a wedges partition. Both have an equal importance in the final signature. Shapes are characterized with a histogram inspired by the Hough transform [Ferecatu2005]. For each pixel, the gradient orientation and the projection amplitude of the pixel vector onto the tangent vector to the local edge are used to build a 2D histogram. Another generic shape descriptor used in IKONA is the 'local edge orientation histogram' which is an extension of the standard edge orientation histogram with some locality constraints.

An example of similarity query using a combination of IKONA global visual features is illustrated on Figure 1.

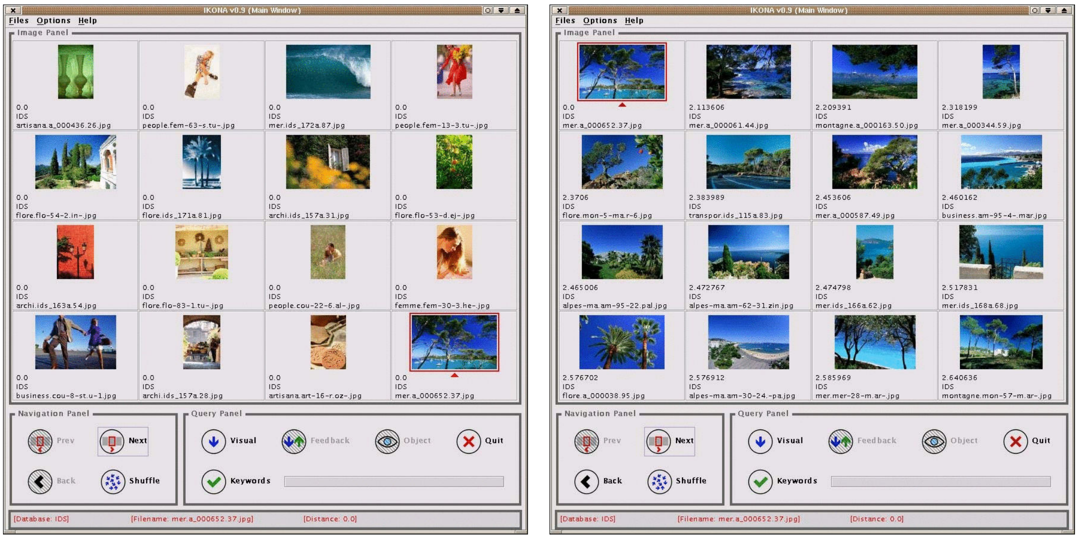

IKONA v0.9 (Main Window)
Files Options Help
Image Panel
0.0 IDS artisana.a_000436.26.jpg
0.0 IDS people.fem-63-s.tu-.jpg
0.0 IDS mer.ids_172a.87.jpg
0.0 IDS people.fem-13-3.tu-.jpg
0.0 IDS flore.flo-54-2.in-.jpg
0.0 IDS flore.ids_171a.81.jpg
0.0 IDS archi.ids_157a.31.jpg
0.0 IDS flore.flo-53-d.ej-.jpg
0.0 IDS archi.ids_163a.54.jpg
0.0 IDS flore.flo-83-1.tu-.jpg
0.0 IDS people.cou-22-6.al-.jpg
0.0 IDS femme.fem-30-3.he-.jpg
0.0 IDS business.cou-8-st.u-1.jpg
0.0 IDS archi.ids_157a.28.jpg
0.0 IDS artisana.art-16-r.oz-.jpg
0.0 IDS mer.a_000652.37.jpg
Navigation Panel
Prev
Next
Back
Shuffle
Query Panel
Visual
Feedback
Object
Quit
Keywords
[Database: IDS] [Filename: mer.a_000652.37.jpg] [Distance: 0.0]
IKONA v0.9 (Main Window)
Files Options Help
Image Panel
0.0 IDS mer.a_000652.37.jpg
2.113606 IDS mer.a_000061.44.jpg
2.209391 IDS montagne.a_000163.50.jpg
2.318199 IDS mer.a_000344.59.jpg
2.3706 IDS flore.mon-5-ma.r-6.jpg
2.383989 IDS transpor.ids_115a.83.jpg
2.453606 IDS mer.a_000587.49.jpg
2.460162 IDS business.am-95-4-.mar.jpg
2.465006 IDS alpes-ma.am-95-22.pal.jpg
2.472767 IDS alpes-ma.am-62-31.zin.jpg
2.474798 IDS mer.ids_166a.62.jpg
2.517831 IDS mer.ids_168a.68.jpg
2.576702 IDS flore.a_000038.95.jpg
2.576912 IDS alpes-ma.am-30-24.-pa.jpg
2.585969 IDS mer.mer-28-m.ar-.jpg
2.640636 IDS montagne.mon-57-m.ar-.jpg
Navigation Panel
Prev
Next
Back
Shuffle
Query Panel
Visual
Feedback
Object
Quit
Keywords
[Database: IDS] [Filename: mer.a_000652.37.jpg] [Distance: 0.0]

Figure 1: Image retrieval by global similarity (the query image is highlighted on the left).

2.1.2 Local features

A wide range of recent image retrieval technologies, and particularly those based on object class recognition, are based on local features [Fergus 2005] [Opelt 2004] [Sivic 2005] [Amores2006] [Datta2006] [Yang2006]. The visual content of an image is not described by a single descriptor but rather by a set of local descriptors describing different regions of the image. This precise description allows reaching better results than global features since small objects of the image can be described separately and can thus be retrieved or learned more accurately. All types of local descriptors capture the various kinds of information encoded in the pixels of one region. As such, there are quite many approaches and different formalisms to describe the same general properties, namely colour and spatial information.

The drawback of these approaches is that they produce a significant larger volume of data, from 10 to 100 times larger than global features. Sometimes, the amount of produced data for a single image can be as large as the compressed image itself. For such approaches the use of efficient access methods and indexing structures is essential in contrast to global features that can often be processed with a single sequential scan of the data (see section 5).

Generally, local feature extraction algorithms perform one or more of the following tasks:

- Localisation of interest points
- Definition of support regions (windows/blobs/segments)
- Extraction of features in the support regions

Each of these tasks could be a part of the proposed algorithms or processed independently by external dedicated algorithms and then combined in a picture workflow.

2.1.2.1 Interest points and support regions

The first step in calculating most local descriptors is the localisation of ‘interest points’, these being points that have some properties suitable for the descriptor in use or the application. Even though they are referred to as points, the term ‘interest region’ may be better suited, since the points are also simply a support region, i.e. a well defined and algorithm dependent spatial extent. Generally, appropriate regions or points are the ones whose neighbourhoods exhibit sufficiently distinctive appearance (the signal changes 2-dimensionally) and/or are invariant to some kind of transformation. One of the first and still widely used detectors of the first kind is the Harris corner detector [Harris1988], proposed back in 1988. It computes the locally averaged second-moment matrix computed from the image gradients, and then combines its eigenvalues to compute a corner “strength”, of which maximum values indicate the corner positions. Instead of explicitly computing the eigenvalues, the strength of a corner is measured as a function of the determinant and the trace of the moment matrix, leading to very easy and fast computation while also being invariant under rotations. Although strong Harris corners indicate regions with spatial change in both directions, which is a prerequisite for most texture or shape based descriptors, they are not invariant to illumination changes and general transformations.

In their seminal works, Witkin [Witkin1983] and Koenderink [Koenderink1984] proposed to handle transformations corresponding to scale change by representing image structures at different scales in a so-called scale-space representation. Basing on this, Lindeberg introduced the concept of automatic scale selection [Lindeberg1988], allowing to detect interest points in an image, each with their own characteristic scale. Interest functions are defined as non-linear combinations of Gaussian γ-normalised derivatives. The chosen interest function is then computed over different scales of the image and local maxima over these scales are extracted. He experimented with both the determinant of the normalised Hessian matrix as well as the normalised Laplacian as interest functions to detect blob-like structures.

Mikolajczyk and Schmid refined this method, creating robust, rotation and scale-invariant feature detectors with high repeatability, which they coined Harris-Laplace and Hessian-Laplace [Mikolajczyk2001]. The idea behind their algorithm is to combine the good localisation properties and rotation invariance of the Harris function, with the scale selection properties of the Laplacian. Deviating from previous approaches which stressed on maximising a single function over the combined 3D scale space, they used a scale-adapted Harris function to compute interest points in 2D for each scale. They subsequently kept only the points that were also local maxima of the Laplacian in scale space. In their subsequent works [Mikolajczyk2002, Mikolajczyk2004], they further extended their method to detect affine invariant interest points. Starting with the scale invariant interest points described earlier, they based on the work of [Lindeberg1994, Baumberg2000] to iteratively compute an affine adaptation matrix around the point neighbourhood, using convolutions with non-symmetric Gaussian kernels. This matrix defines the transformation that converts an elliptic patch around the point to a circular one with isotropic texture, thus detecting up to a rotation affine invariant regions. These patches, along with their characteristic scale, are the support regions for the subsequent calculation of affine invariant features.

Focusing on speed, Lowe [Lowe1999] approximated the Laplacian of Gaussian (LoG) by a Difference of Gaussians (DoG) filter, as a first step to detect scale invariant keypoints for his SIFT algorithm at local maxima in scale-space. The input image is smoothed with the Gaussian kernel of a fixed size. The smoothing is repeated a second time with the same filter. The first level of the DoG representation is obtained by subtracting these two smoothed images. Next, the twice smoothed images are sampled with the scale factor corresponding to the scale of the kernel. The resulting sampled image is used to build the next DoG scale level. All the resolution levels are constructed by combined smoothing and sampling. The local 3D extrema in the pyramid representation determine the localisation and the scale of interest points. The DoG operator is a close approximation of the LoG function but this implementation permits a considerable acceleration of the computation process. In addition this descriptor is inherently invariant to scale changes, resulting in the same values for the same structures in different image sizes.

Lately, Speeded-Up Robust Features (SURF) has been proposed by Bay, Tuytelaars and Van Gool [Bay2006] that further speeds the interest point detection process using a Fast-Hessian detector. In their approach they compute the location and scale of interest points simultaneously, using as a function to be maximised the determinant of a scale-normalised Hessian matrix. The Hessian matrix is in turn approximated by the convolution of the image with 2D Haar wavelets or box filters, which are simple approximations of Gaussian second order derivatives.

Another type of scale-invariant interest point detector is the salient region detector proposed by Kadir and Brady [Kadir2001] that measures the entropy of pixel intensity histograms computed for elliptical regions to find local maxima in affine transformation space. They use an approach inspired from information theory to detect salient points in images. Their work is an extension to the work of Gilles [Gilles1998], who focused on aerial photographs that usually exhibit more modest scale changes and transformations due to the very small relative scale of the objects compared to the depth. Salient points are defined as the centre of regions with high entropy, or from an information theory point of view, regions that have high information content. In order to define entropy, the a priori probabilities of the symbols, in this case greylevel values, are needed. This means that the a priori probabilities describe the model of non-interesting regions and salient points the ones whose attached regions deviate the most from this model. The entropy is also weighted by the sum of absolute difference of the probability density functions of the local descriptor in neighbouring scales, in order to exclude regions that are self-similar in a range of scales and provide unstable salient points. One problem so far is that noise also increases entropy and some more or less flat regions may be selected as salient ones. To overcome this, a simple nearest-neighbour clustering algorithm is finally applied to reject isolated points that are more likely to have been affected by noise.

Baumberg [Baumberg2000] proposes a scale and affine invariant interest point detector that detects spatial Harris features at a set of scales and orders these features based on a scale-normalised feature strength, thus bypassing scale-space extrema detection. The corner strength is defined as a function of the determinant and the trace of the second moment matrix as defined by [Lindeberg1994]. This means that corner strengths can be compared across different scales and the top corners over all detected scales can be determined.

Another variety of algorithms computes features based on contours detected on the image, a field that has regained some interest recently after falling out of favour in the 90s. For these algorithms there exist interest point detectors that select regions suitable for edge-based calculations. In [Mikolajczyk2003], Mikolajczyk et al. define salient edge points as the maximisation of a scale-normalised Canny edge detector function for localisation and the maximisation of the Laplacian for scale selection. The edge-based region detector proposed by Jurie et al. [Jurie2004], is computed over scales and space salient local space convexities. These are regions defined by contours that more or less follow circles centred in the patch. To achieve this, they devise a saliency metric for circular regions as a product of two terms, the tangent edge energy which measures the extent to which the detected edges are strong and well-aligned, with the circle and the contour orientation entropy which measures the extent to which the circle has support from a broad distribution of points around its boundary (and not just from a few points on one side of it).

Matas et al. [Matas2002] introduced Maximally Stable Extremal Regions. In their approach, they adapt the watershed segmentation algorithm for the purpose of interest region detection. Starting from local intensity minima and/or maxima, connected regions are grown over ascending/descending pixel intensity values. Intensity levels that are local minima of the rate of change of the connected area are selected as thresholds producing maximally stable extremal regions. By diagonalising the regions' covariance matrices, affine invariant interest regions are detected.

Tuytelaars and Van Gool [Tuytelaars2002, Tuytelaars2004] construct two types of affine-invariant regions, one based on a combination of interest points and edges and the other one based on image intensities. In the first approach they take into account parallelogram-shaped regions each defined by a Harris corner and points along its edges, calculated using the Canny edge detector or by using local extrema of image intensity as proposed in [Matas2002]. The points stop at positions where some photometric quantities of the texture covered by the parallelogram go through an extremum. A few functions are proposed and it is suggested that many of them should be combined to ensure that a high number of corners indeed generate some regions. The functions are selected so that affine and rotational invariance is achieved. Their second method selects regions defined by function maxima, along rays emanating from local intensity extrema. The functions depend on the intensity values and care is taken that they remain invariant under affine transformations.

A class of methods uses another approach for the detection of interest points, namely wavelets, which are closely related to scale-space theory. Sebe et al. [Sebe2000,Tian2001], describe a scheme that iteratively selects the highest wavelet coefficients of the same region from coarser to finer scale. In [Tonnin2004], Tonnin and Gross use wavelets and curvelets as convolution kernels to detect interest points with point and edge continuities accordingly. They also extend the notion of Mallat [Mallat1989] for the selection of salient points by extending the search for maxima over scales, using a normalisation factor.

Most of the aforementioned interest point detection methods use only the luminance information of the images. Yet, as shown in [Stottinger2007, Montesinos1998], improved performance in retrieval tasks can be achieved by the use of colour information in interest point detection. Based on the Harris corner detector, they explore a new way to use multi-channel images and to select colour scale and evaluate different colour spaces.

Other approaches, and especially those that apply global feature extraction methodologies on local features, require the segmentation of the image. They thus use one of the many segmentation algorithms to select regions as salient ones. In [Fauqueur2004], Fauqueur et al. use the Competitive Agglomeration classification algorithm to determine a set of characteristic quantized colours. For each pixel, local distribution is determined on the set of quantized colours providing a LDQC feature (Local Distribution of Quantized Colours), which are then grouped by the same classification algorithm to generate coarse regions.

Other methods choose interest points using criteria inspired from human vision system, symmetry properties, which are known to catch well visual attention. Loy and Zelinsky [Loy2003] introduced a transform developed to detect radial symmetry centres. Generalising on their findings, Rebai et al. [Rebai2006] extended this algorithm to a set of voting orientations which allows the construction of a more complete analysis space, enabling the detection of points with different topological natures with the same detector.

Detailed comparisons of interest point detectors and evaluations on benchmarking datasets can be found in [Sebe2002, Mikolajczyk2005a, Mikolajczyk2005b]. However, it should be noted that there is no rigorously established framework for the evaluation of interest point detectors. Most used criterion is the stability of the detected patterns after photometric or geometric transformations of the image. This is very useful for invariance concerns but it does not evaluate the visual relevance of the detected patterns. Other criteria try to estimate the information content of the local region but this can only estimate the quality of the detector alone since it depends on the choice of a local descriptor. That's why interest point detectors may be chosen and evaluated according to the local features that will be used to characterize the local content.

2.1.2.2 Local feature descriptors

A very large variety of feature descriptors have been proposed, that work either independently from the interest point localisation or not. The main task is to adequately describe the support regions using robust geometric and photometric information. Each technique provides different levels of robustness, both regarding noise and image transformations and as such many are application specific. Generally, the feature descriptors may incorporate quantities calculated by the interest point detector, or ignore them and extract information only from the support regions. A part from this two different approaches can be used to describe an image from a number of salient points or regions, a global and a local one. In the former case a signature for the image is formed by considering the points globally, while in the latter a signature is extracted from each individual one. The global approach relates to the methodologies of classical global feature description schemes and will not be discussed here. As an example, in [Sebe2000], whose salient point selection mechanism using wavelets is described above, colour distribution moments are calculated from salient regions and used as a global descriptor. Many of the descriptors of all three categories described below either coincide or are inspired by the shape and colour descriptors of MPEG-7. [Salembier2002, Martinez2002]

The simplest descriptor is composed of the pixel values themselves, however for complexity reasons it is seldom used, primarily for finding correspondences between small baseline images. More commonly, feature description algorithms transform the pixel values to some other domain, resulting in the following classification for the algorithms.

2.1.2.2.1 Distribution-based descriptors

These descriptors use histograms to represent various characteristics of appearance or shape, such as colour and luminance, edge and gradient orientations, radial distances and texture. A simple descriptor is the distribution of the pixel intensities represented by a histogram. A more expressive representation was introduced by Johnson and Hebert [Johnson1997] for 3D object recognition in the context of range data. Their representation (spin image) is a histogram of the point positions in the neighbourhood of a 3D interest point. This descriptor was also adapted to images [Lazebnik2003]. The two dimensions of the histogram are distance from the centre point and the intensity value. A related method that uses the maxima of the intensity values along rays emanating from the interest points has been described in [Tuytelaars2000]

Zabih and Woodfill [Zabih1994] have developed an approach robust to illumination changes. It relies on histograms of ordering and reciprocal relations between pixel intensities which are more robust than raw pixel intensities. The binary relations between intensities of several neighbouring pixels are encoded by binary strings and a distribution of all possible combinations is represented by histograms. This descriptor is suitable for texture representation, but a large number of dimensions is required to build a reliable descriptor [Ojala2002].

Shape context [Belongie2002] computes a histogram describing edges distribution in order to characterize the shape of the local patterns. Edges are extracted by the Canny detector and location is quantized into bins of a log-polar coordinate system. This method has been successfully used in cases for which edges are very stable which is not the case in general. Carmichael & Hebert [Carmichael2003] take a similar approach, characterising each edge pixel by the local distribution of edges in its image neighbourhood. On similar grounds Yahiaoui et al. [Yahaoui2006] propose a shape descriptor, called Directional Fragment Histogram (DFH), constructed by approximating contours by quantized directional segments, whose length and direction is used to construct the DFH.

Lowe [Lowe2004] proposed a Scale Invariant Feature Transform (SIFT), which combines a scale invariant region detector as described before and a descriptor based on the gradient distribution in the detected regions. The descriptor is represented by a 3D histogram of local oriented gradients around the interest point weighted by the gradient magnitude and stored in a 128-dimensional vector (8 orientation bins for each of the 4 × 4 location bins). The rich information on gradient locations and orientations makes the descriptor robust to small geometric distortions and small errors in the region detection.

Various refinements on this basic scheme have been proposed. Ke and Sukthankar, for example, [Ke2004b] applied PCA to reduce the dimensionality of SIFT features. This PCA-SIFT yields a 36-dimensional descriptor which allows fast matching. In [Micolaczyk205a], Micolaczyk and Schmid have proposed a variant of SIFT, called GLOH (Gradient Location-Orientation Histogram), designed to increase its robustness and distinctiveness. The SIFT descriptor is computed for a log-polar location grid with three bins in radial direction and 8 in angular direction (excluding the central bin), which results in 17 location bins. The gradient orientations are quantised in 16 bins. This gives a 272 bin histogram, reduced by PCA to 128 values.

Associated to their interest point detector, Bay et al. [Bay2006] also proposed new scale- and rotation-invariant local features coined SURF. The first step consists of fixing a reproducible orientation based on information from a circular region around the interest point. Then, they construct a square region aligned to the selected orientation, and extract the SURF descriptor from it, thanks to structured distributions of Haar wavelets coefficients.

Szumilas et al. [Szumilas2007] focus their ORC descriptor (Orientation-invariant Radial Configuration) on detecting more perceptible local structure than what is detected with SIFT. They extract feature centre locations at places where a symmetry measure is maximised. Next, boundary points along rays emanating from the centre are extracted. Boundary points are defined as edges or transitions between relatively different regions, and are extracted by hierarchical clustering of pixel feature values along the ray. Grouping of boundary points is also performed between adjacent rays. The ideas employed in this method focus on capturing the appearance of naturally incurring object patches and on selecting more perceptible features localised on them.

Fauqueur et al. [Fauqueur2004] devise a new histogram based colour descriptor that uses distributions of quantised colours, previously employed in global image feature techniques, in the local feature extraction case. Considering that description must be finer for regions than for images they propose region descriptor of fine colour variability: the Adaptive Distribution of Color Shades (ADCS). They combine ADCS with an appropriate similarity measure to enable its use in indexing.

2.1.2.2.2 Differential and moment descriptors

These descriptors incorporate information related to the derivatives of the variation of the image in the neighbourhood of the interest points. These variations either correspond to the luminance or colour of the image or to changes of the local structure such as edges and contours. Alternatively moment invariants are used to model the structure of regions.

A set of image derivatives computed up to a given order approximates a point neighborhood. The properties of local derivatives (Local Jet) were investigated by Koenderink and van Doorn [Koenderink1987]. Florack et al. [Florack1994] derived differential invariants, which combine components of the Local Jet to obtain rotation invariance. The components of the Local Jet can be combined in various ways to obtain invariance to rotation, affine transformations and photometric changes, as shown in [Gool1996]. A generalisation to color images has been proposed in [Montesinos1998], allowing for the description with less sensitive to noise lower order derivatives compared to grey-valued images.

Freeman and Adelson [Freeman1991] developed steerable filters, which steer derivatives in a particular direction given the components of the Local Jet, thus making them invariant to rotation. A stable estimation of the derivatives is obtained by convolution with Gaussian derivatives. Shi et al.[Shi2006] introduce a technique to design general rotational invariant operators based on steerable filter banks that can be used for local features in texture analysis.

Baumberg [Baumberg2000] and Schaffalitzky and Zisserman [Schaffalitzky2002] proposed using complex linear filters to extract rotational invariants. For the filters, Baumberg uses a variant of the Fourier-Mellin transformation to calculate a set of complex-valued coefficients for each colour component, after applying a transformation that diagonalises the regions covariance matrix of the detected regions which in combination is an affine-invariant procedure. He also experimented with rotational and affine invariant colour moments, introduced in [Mindru1992], with similar results. On the same grounds, Schaffalitzky and Zisserman apply a complex polynomial, differing from the Gaussian derivatives by a linear coordinates change in the filter response domain.

Another approach similar to the ones described above by Carneiro and Jepson [Carneiro2002], uses phase information as a local feature. Their phase-based local feature is a complex representation of local image data that is obtained through the use of quadrature pair filters, tuned to a specific orientation and scale. In their subsequent work they extended their algorithm to handle multiple scales [Carneiro2003], by checking local spatial information to determine whether the current scale is appropriate.

Invariant features under some transformations can also be expressed in terms of generalised colour moments. Mindru et al. [Mindru2004] explore such moment invariants for several combinations of deformations and photometric changes. They also offer a concept through which they can also be used to deal with the recognition of non-planar 3D objects.

Finally, a variety of algorithms uses the Curvature Scale Space, constructed by considering the derivative of the tangent angle to image curves at different scales, to define shape-based descriptors for image regions. Usually, the image is segmented at first or its edges are extracted. Mokhtarian et all. [Mokhtarian1996] use the maxima of curvature zero crossings in Curvature Scale Space to describe object boundary contours.

2.1.2.2.3 Spatial-frequency descriptors

These descriptors model spatial frequency in the vicinity of interest points, as expressed by transforming the 2D intensity value field to a set of frequency coefficients obtained by different transformation kernels (Fourrier, DCT, Gabor, Wavelets)

Many techniques describe the frequency content of an image. The Fourier transform decomposes the image content into the basis functions. However, in this representation, the spatial relations between points are not explicit and the basis functions are infinite; therefore, it is difficult to adapt to a local approach. The related Discrete Cosine Transform has found many applications as a feature descriptor because of its decorrelation properties.

The Gabor transform overcomes the problems of compactness of the Fourrier transform, but a large number of Gabor filters is required to capture small changes in frequency and orientation. An added value to Gabor filters is that they seem to simulate some aspects of pattern recognition present in the human vision system and have been often used in face recognition methods. Moreno et al [Moreno 2006] propose scale and rotation invariant models based on Gabor functions to be used in interest point selection and classification. For a comparison of texture features based on Gabor filters see [Grigorescu2002].

Lastly, the wavelet transform [Mallat1989] has been frequently explored in the context of texture classification, though rather in specialised applications like iris and character recognition.

2.1.2.2.4 Comparison and analysis

Comparison of some of the abovementioned feature descriptors and evaluations on benchmarking datasets can be found in [Mikolajczyk2004, Mikolajczyk2005a, Deselaers2004]. SIFT descriptor [Lowe2004] and its variants [Ke2004b] [Micolaczyk205a] have been shown to often outperform the others. This can be explained by the fact that they capture a substantial amount of information about the spatial intensity patterns, while at the same time being robust to small deformations or localisation errors. The SIFT descriptor still seems to be the most appealing descriptor for practical uses, and hence also the most widely used nowadays. It is distinctive and relatively fast, which is crucial for on-line applications. Recently, Se et al. [22] implemented SIFT on a Field Programmable Gate Array (FPGA) and improved its speed by an order of magnitude. However, the high dimensionality of the descriptor is a drawback of SIFT at the matching step. For on-line applications on a regular PC, each one of the three steps (detection, description, matching) have to be faster still. Lowe proposed a best-bin-first alternative in order to speed up the matching step, but this results in lower accuracy.

The SURF descriptor is based on similar properties, with a complexity stripped down even further and it seems to outperform SIFT descriptor and its variants both in speed and accuracy. It is mixing of crudely localised information and the distribution of gradient related features seems to yield good distinctive power while fending off the effects of localisation errors in terms of scale or space. Using relative strengths and orientations of gradients reduces the effect of photometric changes.

However, SIFT like descriptors or SURF descriptors still encounter some problems:

- They suffer from a lack of visual relevance and the amount of generated data makes their use really problematic for large multimedia databases. It is known that, for generic object recognition techniques, only a small percentage of initially extracted SIFT features are used in the final object model. This proves that most of these features are useless and unsuitable for such tasks. This constitutes a problem both in terms of quality and computation time. These points correspond to stable and informative signal properties but they do not systematically coincide with human visual criteria.

- The second problem of such descriptors is that they still do not solve the problem of visual shapes description. They include a certain degree of structural information but not enough to recognize real visual shapes.

Due to these two remarks, we think that the design of efficient and more visually relevant local features is still an issue.

2.1.3 Video features

All the previously described techniques can of course be applied to describe the visual content of video frames and are in fact widely used. If we look at TRECVID benchmark as reference, we can see that most of the techniques are indeed based only on frame descriptions techniques to extract visual concepts. During the 2006 campaign, only 8/30 teams look at more than just the shot keyframe. This is due to several reasons:

- Motion based features are usually less stable and more affected by compression.
- Computational costs induced by motion estimation are high.
- Few of the generally targeted concepts are based on motion properties.

In VITALAS Projects, we foresee the following applications that handle videos:

- Near Copy Detection: Contrary to popular belief, a copy is not an identical document or a near replicated document but rather a transformed document, i.e. a document that has been obtained from an original one by a succession of processes. The process of copy detection is to retrieve the copy of a video candidate in a catalogue or a database of videos.
- Sets retrieval: Settings are time and place in which the video's story takes place, can also emphasize atmosphere or mood. Sets retrieval has the aims to identify the programmes that have the same settings. The programmes with the same settings are supposed to belong to the same collection or series of programmes. This matching can generate content links can determine where and when a document has been broadcasted. This kind of information allows to generate high level semantic metadata.
- Pattern retrieval : Static or dynamic entities in the video document, which can be a logos, an object, a human being, etc. The retrieval of these entities can for instance help to localise an event or a type of programs.

As still image features are described in the previous section, we will not provide many details about key frames based techniques but rather focus on methods that can be useful for our applications' objectives. We will then focus on some recent scalable techniques based on temporal and dynamic information that could be used in the scope of VITALAS.

2.1.3.1 Key-frames strategies

In [Sivic2006], two types of elliptical affine co-variant regions are used to describe the image contents of key-frames. The implementation details are given in [Mikolajczyk2002, Schaffalitzky2002]. Each elliptical affine co-variant region is represented by a 128-dimensional vector using the SIFT descriptor developed by Lowe [Lowe2004]. The combination of SIFT descriptor with affine covariant regions provides region description vectors which are invariant to affine transformations of the image.

Instead of introducing directly these descriptors in an index structure, they are post-processed in order to build a "visual vocabulary". The objective of this post-processing is to vector quantize the descriptors into clusters. Each cluster represents the visual words as for text retrieval. The number of clusters is chosen to maximize retrieval results on a ground truth data. Each descriptor for a new frame of the movie are associated to visual words. The associated visual word is the nearest cluster centre to their SIFT descriptor.

The final representation of video is a set of key frames, and each key frame is represented by the visual words and their position. This representation allows to introduce the information in an inverted files [Baeza1999].

This strategy allows to retrieve objects in videos. The objects can be affected by very strong transformations (zoom, rotations…). The use of inverted files provides low computational costs in order to search descriptors. The building of the visual vocabulary can be a complex task to be adapted for a large set of videos. This type of technologies could be efficient to track objects, sets and people in a video or a set of videos.

In [Joly2004] and [Joly2007], the used local features are based on an improved version of the Harris point detector and a set of local jets computed in the neighbourhood of each interest point. To increase the compression, the features are not extracted in every frame of the video but only in key-frames corresponding to extrema of the global intensity of motion [Eickeler1999]. Since, the information content of the stable Harris detector [Harris1988] is high for differential features; they use a Gaussian differential decomposition of the gray level 2D signal until the second order. The differential decompositions are combined to be invariant to illumination variations and geometric transformations. In order to include temporal information, the feature vector is also computed at three other instants around the current key-frame but not at the same position to avoid redundancy in the frequent case of the still scenes. The final local features are 20-dimensional vectors in [0, 255] and the mean rate is about 17 local features per second of video. This type of descriptors has a limited robustness for scale changes and is not invariant to rotation. However, they have the advantage to be very discriminative, robust to localization errors and require a low computational cost. This makes them very efficient for content-based copy detection tasks. The technique fits well for hundreds of thousands hours of videos and is then very adapted for the scalability issue.

2.1.3.2 Spatio-temporal interest points

Instead of detecting points of interest in key-frames, Ivan Laptev and Tony Lindeberg [Laptev2003] had extended the concept of points of interest in the spatiotemporal space. They transform the Harris and Forstner interest point operators and detect local structures in space-time. They measure the spatiotemporal extents of the detected events and associate scale-invariant spatiotemporal descriptors.

These spatiotemporal points of interest have been combined with different types of descriptors:

- N-jet of order 4 at a single scale [Laptev2003],
- Multi-scale N-jet of order 4 [Laptev2003],
- Local position dependent histograms of first-order partial derivatives,
- Local position independent histograms of optic flow,

- Local principal component analysis of optic flow,
- Local principal component analysis of spatio-temporal gradients vectors,

This combination has been used to recognize types of activities [Laptev2004] and applied to video copy detections [Law2007].

Spatiotemporal points have the advantage to characterize very salient patterns and are thus adapted to describe temporal events. Moreover, this saliency is very efficient for the video copy detection.

However, such method makes difficult the process of very large catalogues of videos, because the computational costs are very high. This detector requires at least 3 convolutions in the 3 dimensions.

2.1.3.3 Interest points trajectories

In [Moenne2006], [Law2006], they propose a new description of video sequences, which is based on the dynamic content.

In [Moenne2006], the visual description relies on SIFT trajectories of the video sequences. The set of SIFT (invariant local features) are detected for every frames of a video sequence. Trajectories are then estimated by matching the features of successive frames. This method is based on the classical algorithm of Tomasi [Shi1994]. The similarity measure is then not based on the visual content of the images on a representation of a set of trajectories. This representation is equivalent to the Visual Vocabulary. The trajectory space is quantized with a vocabulary. This vocabulary is composed of motion models that are trajectory features. The descriptor is then a histogram that is representing the number of occurrences of each local motion model. This type of descriptors has been developed to recognize activities (Boxing Jogging Walking Running HandWaving HandClapping).

In [Law2006], the author developed an equivalent indexing strategy but using two different kind of local features (Local jets around Harris points and Symmetry points) and an asymmetric feature extraction strategy. For the indexing stage, the features are computed in each frame and tracked to build the trajectories. However, the trajectory is not used directly as local feature but simply used to compute a bounding box saved as metadata. The feature itself is based on the local visual content only, by averaging the local jets along the trajectory. During the search stage the visual features are computed only in key-frames and the trajectories are not constructed. Then, a specific registration algorithm allows to match the points' positions of the queries with the indexed trajectories. This asymmetric strategy drastically limits the computational costs of the search process. It also provides an accurate and flexible tuning during the search process: the rate of extracted local features can be changed on-line, and then the granularity of the targeted video segments can be adapted to the application.

Interest point trajectories are efficient to describe the dynamic contents of the video sequences. Moreover, the dynamic information is estimated with the trajectories of the interest points that allow compressing temporal information, contrary to the method founded on the dense optic flow. However, this strategy requires computing the interest points on each frame of video sequences, which induces high computational costs. This could be a bottleneck when the aim is to process a very large catalogue of videos.

2.1.4 Similarity measures

Once a decision on the choice of visual signatures is made, how to use them for accurate image or video retrieval is the next concern. There have been a large number of fundamentally different frameworks proposed in the recent years. Some of the key motivating factors behind the design of the proposed visual similarity measures can be summarized as follows:

- Agreement with semantics

- Robustness to noise (invariant to perturbations)

- Computational efficiency (ability to work real-time and in large-scale)

- Invariance to background (allowing region-based querying)

- Local linearity (i.e., following triangle inequality in a neighbourhood)

The various techniques can be grouped according to their design philosophies, as follows:

- treating features as vectors, non-vector representations, or ensembles

- using region-based similarity, global similarity, or a combination of both

- computing similarities over linear space or non-linear manifold

- image segments based similarity computation

- stochastic, fuzzy, or deterministic similarity measures

- use of supervised, semi-supervised, or unsupervised learning

We will not detail in this document the huge set of different similarity measures since there already exist very good categorizations, e.g. in [Smeulders2000] and [Datta2007]. Distances for comparing global features are notably clearly detailed and compared in tables. A discussion regarding the usability of global similarity measures in index space structuring techniques will be given in section 5 of this document. In the following section, we provide a state-of-the-art of local features matching techniques since some of them are more recent and less popular.

2.1.4.1 Local features matching

Each local feature extracted from an image or a video provides two types of information: The *location* of the corresponding interest point, as well as its *descriptor*. Both localization and description vary, depending on the method applied. A distance metric is usually given that can measure the similarity between two interest point descriptors. However, many features extracted from an image will not have any correct match in another image, because they arise from background clutter or were not detected in both images (or videos).

In order to efficiently utilize local features for image/video retrieval or object recognition tasks, one must devise robust techniques for local features matching that can select the correct interest points while rejecting the "outliers". Matching can be performed with or without setting any spatial constraints on the interest points to be matched (i.e., take the interest point location into account or not). Many methods presented in the literature actually use both; an initial step finds candidate matches without the use of spatial information, while a second step rejects incorrect matches based on spatial information.

2.1.4.1.1 Matching without spatial registration

In the simplest case, given two images, I_1 and I_2, and their interest points setsF $\mathbf{P}_1$ and $\mathbf{P}_2$, the nearest neighbour q of each $p \in \mathbf{P}_1$ is selected from $\mathbf{P}_2$ based on the descriptor distance. If the distance between p and q is below a predefined global threshold, it is considered as a correct match. As D. G. Lowe stated in [Lowe2004], this approach does not perform well, since some descriptors are more discriminative than others. He observed that for a false match it is likely that there will be a number of other false matches within similar distances due to the high dimensionality of the descriptors used. Thus, a more effective measure, called Nearest Neighbour Distance Ratio (NNDR), is obtained by comparing the distance of the closest neighbour to that of the second closest neighbour. In the reported experiments, Lowe indicates that thresholding applied to the ratio of the nearest to the second nearest match of an interest point eliminated 90% of the false matches, while discarding less than 5% correct matches.

Nearest Neighbour (NN) and NNDR were both used for local feature evaluation purposes in [Mikolajczyk2005a], where it was shown that: (i) The NNDR approach generally performs better than NN and (ii) the performance of interest point detectors changes depending on the matching strategy used. Both NN and NNDR have been used in the literature for a variety of matching problems. [Mikolajczyk2004] used an NN matching strategy with thresholding as an initial matching step in their detector, followed by a verification step based on the cross-correlation of affine normalized image patches. [Bamberg 2000] used NNDR as an ambiguity measure and selected the n most “unambiguous” matches to solve the uncalibrated wide baseline stereo problem, while NNDR was also used in [Bay2006] for evaluation of the SURF descriptor.

In [Brown2005] the authors observed that for matching interest points from multiple images (i) the distance of the second best match (and subsequent matches) remains almost constant in feature space and (ii) better outlier discrimination is achieved if the average of the second best match is obtained from multiple images. [Tuytelaars2004] introduced an additional photometric constraint for further rejection of false matches. It is assumed that some parts of the images to be compared have similar illumination conditions. A photometric transformation is assumed, that includes scaling and offset of the RGB colorspace. For two matching regions (around the interest points), the scale factors and offsets of the photometric transformation are computed using moments. Then, given a pair of region correspondences (two distinct matches), their photometric consistency is computed by comparing their photometric transformations. A match is discarded as incorrect if it is consistent (based on a threshold) with less than n correspondences.

All the methods indicated so far can be used to reject false matches while keeping the correct matches of local features computed between two images. However, they do not solve the original retrieval problems (image matching / object recognition). In most cases, a voting scheme is employed, often involving a clustering or trained classification procedure, in order to determine whether two images match, or whether an object can be recognized in an image. The need for such an approach was highlighted in [Lowe2004] where it was noted that for small or highly occluded objects, there is a need to perform identification with the fewest possible number of matches.

In [Wallraven2003] the use of Support Vector Machines (SVMs) is proposed for matching local features. But since local representations generally consist of feature vectors of different length there is no structure between the local features and they cannot be directly used with SVMs. The paper proposes the use of a new class of SVM kernels for use with local features and provides a proof that this kernel satisfies Mercer’s condition. As noted in [Boughorbel2005], however, this proof is not correct, even though in the experiments reported the accuracy is good. The authors in [Boughorbel2005], propose the intermediate matching kernel for local feature matching. The method proposes the use of fuzzy C-Means for the construction of a set of virtual local features that are actually the class centers of the clustering of whole local features extracted from training images. A mapping is then defined, that maps the original local feature sets to R^d, based on a virtual local features. Using such mappings, the intermediate matching kernel is defined and is proved that it is positive definite. A different kernel – based matching approach is presented in [Grauman2005b] that maps feature sets to multi-resolution histograms and computes a weighted histogram intersection in this space. This "pyramid match" computation is linear in the number of features, and it implicitly finds correspondences based on the finest resolution histogram cell where a matched pair first appears. It is also shown that the corresponding kernel satisfies the Mercer condition.

In [Grauman2005a], the authors argue that matching performed by considering each local feature independently ignores useful information captured by the co-occurrence of a set of distinctive image features. The approach proposes the computation of similarity between discrete feature distributions using the Earth Mover's Distance (EMD) that does not require that two images have the same number of local features. An embedding is used that maps an unordered point set to a single point in a L_1 normed space, such that the distance between embedded vectors is comparable to the EMD distance of the feature sets themselves. Thus, instead of using a voting scheme where each local feature contributes independently, feature sets are used collectively.

The combination of SVM and EMD techniques is proposed in [Zhang2006]. Each image is assigned a signature, which is extracted by clustering of local features in the image. The signature is composed of pairs of cluster center and the relative size of the cluster. Two signatures are compared using EMD, without requiring the signatures to cover an identical number of clusters. Then, Kernel-based classification is applied for object recognition and texture classification, where generalized Gaussian kernels are used. The kernel accepts as input the distance between two image signatures and the decision is based on the sum of its value over the training images. Extensive evaluation shows that this approach provides very promising results.

Lepetit et. al. [Lepetit2005] proposed the use of randomized trees for real time keypoint recognition. Again the matching problem is treated as a classification problem solved by decision tree classifiers that are constructed in a top down manner, with the node tests being selected so as to achieve best discrimination based on the training examples. Since producing optimal trees quickly becomes an intractable problem, multiple randomized trees are grown. Rejection of outliers can be performed via thresholding of a confidence measure that is computed during the tree classification process.

A fast method for local feature matching is proposed in [Ke2004a], where it is used for near duplicate detection and sub image retrieval. The method is based on indexing with LSH to return images that have the best number of matches in the image database.

2.1.4.1.2 Matching with spatial registration

The approaches presented so far depend mainly on local feature descriptors for matching, disregarding spatial information (interest point locations and their spatial relations). It is common, however, for methods that originally utilize descriptor similarity to identify correct matches to have an additional processing step that depends on point locations for further rejection of outliers. For example, in [Ke2004a], [Tuytelaars2004], [Mikolajczyk2004] and [Brown2005] presented in the previous section, RANdom SAmple Consensus (RANSAC) is used to improve matching of local features between two images by performing a registration. RANSAC can be used to perform robust estimation of the fundamental matrix or homography between two images. Through an iterative process, the parameters of the matrix (or homography) are estimated, by selecting a subset of the observed interest points as "inliers" and considering the remaining points as "outliers". Thus, the model parameters are computed so that there are sufficient inliers fitted to the model. In [Nister2005], an improved (in terms of computational time) preemptive scoring scheme for RANSAC is proposed, that can lead to faster outlier rejection.

RANSAC can efficiently reject outliers that cannot be filtered out by comparing interest point descriptors, such as those that have similar values, but belong to different objects in the images. But RANSAC (as well as other robust estimation techniques) performs poorly when the number of actual inliers falls much below 50%. Hence such methods are not effective when dealing with background clutter or occlusions. In [Lowe2004] the use of the Hough transform is proposed: The Hough transform identifies clusters of features with a consistent interpretation by using each feature to vote for all object poses that are consistent with the feature. When clusters of features are found to vote for the same pose of an object, the probability of the interpretation being correct is much higher than for any single feature. For each object model, a Hough transform is created that predicts model location, orientation and scale from the match hypothesis, allowing large error bounds (to account for projection and deformation distortions). This transform allows the computation of affine parameters of the observation w.r.t. the model and then the outliers are rejected based on the agreement between each feature and the model. A similar approach is considered in [Platel2005], but instead of applying the Hough transform, clustering of features are computed using the Shared Nearest Neighbor approach (SNN).

In [Lu2004] a different geometric restriction is assumed for the case of the uncalibrated wide baseline problem. It is the "cross – epipolar ordering constraint", that relies on the natural ordering of the epipolar lines that is preserved in both images. The cross – epipolar property contains the set of candidate matches; since the correct matches must be realizable, by rejecting non-realizable matches the likelihood of mismatch due to ambiguity is reduced.

[Lazebnik2006] introduces an extension of the pyramid match kernels presented in [Grauman2005b] so that spatial information is included as well and matching is performed based on approximate global geometric correspondence. The method works by partitioning the image into increasingly fine sub-regions and computing histograms of local features found inside each sub-region. The method was tested with variants of SIFT descriptors and produced high accuracy for scene and object category recognition.

Deng et. al. [Deng2006] proposed the use of local context information as a means to introduce spatial information in local feature matching. For each region around the detected interest point, the dominant gradient orientation is computed and is chosen as a reference orientation of an elliptic region. The detected region is scaled so as to obtain two additional regions that are 8 and 16 times larger than the detected one. Local features within these enlarged ellipses form the "region context" for the center feature. Using normalization of regions and the context of detected features, the matching accuracy is improved. The authors indicate that this matching strategy is more robust and flexible than RANSAC style methods in that it effectively ignores outlier matches without requiring a consistency model for each type of transformation. Another method that utilizes local context information is presented in [Kim2006] that proposes a voting based object recognition method robust to background clutter. The matching is evaluated based on both descriptor distance and proximity with other local feature matches and their descriptor distance, in a weighted voting scheme. Evaluation based on NN and NNDR matching shows that including proximity information in the matching process leads to better results.

In the case of video data, additional temporal information can be included in the matching process. This is the case in [Caspi2006], where interest points are tracked in time and matching is performed on the point trajectories to achieve video sequence synchronization. Feature trajectories are constructed from each sequence and then trajectories are matched (based on multiple point matches). Then both spatial and temporal parameters are computed that allow synchronization of videos of the same scene with different cameras to be synchronized. In [Law2006] for near copy video detection, multiple interest point trajectories are matched from both sequences in an initial processing step. Rejection of false matches is performed using both spatial and temporal registration. Laptev [Laptev2005], on the other hand defined space-time interest points directly for video event detection. Matching is performed based on descriptor distance and clustering of interest points. In [Hess2007] video registration is performed by initially identifying globally distinctive features and calculating for each video frame the homography matrix. Then, using NNDR, locally distinctive features (those that satisfy the NNDR test in an image region only) are identified that in the case of video allow matching of the same features in subsequent frames and to identify additional matches between two videos. The use of locally distinctive features improved the registration accuracy significantly, according to the reported experiments.

2.1.5 Conclusions

Regarding the usefulness of global features, we think that it is essential to integrate some of them in a system such as VITALAS since they are still useful for a wide range of emerging techniques and applications. The global features developed at IMEDIA team, for example, have proven to be very efficient in ImagEval 2006 campaign both for high level visual concept extraction and object recognition. On the other side, the design of efficient features representing well the global visual content of an image can be now considered as a mature technology. We thus will probably not investigate this issue in more depth in the research tasks of VITALAS.

Regarding the use of local features, SIFT and SURF descriptors are now used in a wide range of technologies and they have proven to be very efficient as well as generic. Those kinds of techniques should be considered as baseline algorithm for the VITALAS system. However, these descriptors still face some critical issues: they generate a huge amount of data that is problematic regarding scalability issues and they still do not describe well the visual local shapes of the images. There is thus a need to investigate new geometric primitives and local appearance descriptors in order to produce a smaller but more complete description, representing parts of many object classes with a better visual meaning.

Features really dedicated to video contents are not so numerous and most techniques just use extensions of still image content description. Features based on motion are indeed more time consuming, more unstable under video compression and irrelevant for most targeted visual concepts. However, including temporal information in the content description can improve the performance on tasks requiring high levels of discrimination, such as near-duplicate, logo, objects or sets retrieval. For VITALAS, we suggest to explore the usefulness of dynamic information in more depth.

The choice of a similarity measure depends on many factors such as the feature types, the robustness and invariance requirements and the computational cost. Several global similarity measures and local features matching techniques should be integrated into the VITALAS system in order to satisfy the different needs. It is foreseeable that fast matching techniques with spatial registration will be required for copies, sets and rigid objects retrieval. Different kernels for global features will have to be tested for visual concepts learning. The best similarity measures for the new features developed in VITALAS will have to be defined.

2.2 Objects detection and recognition

2.2.1 Introduction

Although research in computer vision for recognizing objects in photographs dates back to the 1960s, progress was relatively slow until the turn of the millennium, and only now do we see the emergence of effective techniques for recognizing object categories with different appearances under large variations in the observation conditions.

We deal with the problem of detecting the presence or absence of one object or one object category in an image. In contrast to simple object recognition, object-class recognition is not restricted to images of the same physical object (e.g. different images of the same car), but deals with different instances of the object, e.g. images of different cars. This introduces a high variability of appearance across objects of our category. The difficulties increase by the presence of clutter in the images, partial occlusion and accidental conditions in the imaging process.

In the following, we first review the techniques based on bag-of-words models which provided the best performance improvements in the last few years. We then review recent part-based methods that are currently the most studied. Finally we discuss up-to-date evaluation results to give a panorama of most popular and efficient techniques.

2.2.2 Bag-of-words models

Bag-of-Words is a family of methods that treats images as a collection of regions, ignoring their spatial structure. Similar models have been successfully used in the text community for analyzing documents, since each document is represented by a distribution over fixed vocabularies. Although these methods are traditionally used for image classification, several approaches address the problem of object recognition as well.

Early "bag of words" models were used mostly for texture recognition. In [Leung2001], texture is characterized by its responses to a set of orientation and spatial-frequency selective linear filters. Filter responses are then clustered into a small set of prototype response vectors called textons. There are a total of 48 filters (36 elongated filters at 6 orientations, 3 scales, and 2 phases, 8 center-surround difference of Gaussian filters, and 4 low-pass Gaussian filters). In order to capture 3D textures and characterize their local geometric and photometric properties however, filter responses from different pictures of the same material are concatenated and then clustered using the k-means. The resulting centers of the clusters are called 3D textons. Thus, a vocabulary of 3D textons is constructed which is learned with the k-means.

A similar approach is adopted in [Varma2002]. In this work, texture is modeled as a distribution over textons, but in this case, the clustering (k-means) is performed in an extremely low dimensional space as in [Schmid2001] and texture classification is performed from a single image. Lazebnik et al [Lazebnik2003] also use texture images that are modeled as sets of regions. Each region is described by an intensity descriptor that is invariant to affine geometric and photometric transformations, based on spin images, introduced by Johnson and Herbert [Johnson1999]. Clustering is performed on these descriptors in a discriminative way, using a standard agglomerative algorithm. The distribution of the descriptors is then summarized in a form of a signature consisting of cluster centers and relative weights indicating the size of the clusters. Applying the Earth Mover's Distance to signatures, a distance matrix that is used for classification and retrieval is constructed.

Recently, "bag of words" models have made great progress in object categorization. A generative bag of words method was proposed in [Csurka2004]. In this approach, Harris affine points are detected by an iterative process. Firstly, positions and scales of interest points are determined as local maxima (in position) of a scale adapted Harris function, and as local extrema in scale of the Laplacian operator. Then an elliptical (i.e. affine) neighborhood is determined. This has a size given by the selected scale and a shape given by the eigenvalues of the image's second moment matrix. The selection of position/scale and the elliptical neighborhood estimation are then iterated and the point is kept only if the process converges within a fixed number of iterations. SIFT descriptors are then computed in these regions. The model uses either Naive Bayes or multiple SVMs to classify "bag of keypoints" to categories. "Bag of keypoints" is a vector of votes on a predefined (from training) key point list.

Another generative approach to the problem is described in [Blei2004], where each document is described by a set of words and the method tries to discover common usage patterns or "topics" in the documents and to organize these topics into a hierarchy. Hierarchies are tree structures in where each node is associated with a topic, which is a distribution across words. A document is then generated by choosing a path from the root to the leaf. Learning is performed by a combination of hierarchical latent Dirichlet allocation with the nested Chinese Restaurant Process (CRP [Aldous1985]). This method is generic and can be applied to any type of multimedia document.

Finally, in [Wang2006a] images are modeled as distributions over a visual dictionary. The model proposed here is an extension of generative "bag of words" that takes advantage of the interdependency of local image patches via a linkage structure that enforces semantic connection between patches. This is achieved with the use of a variation of the Hierarchical Dirichlet Process (HDP) [Teh2006] called dependent HDP (DHDP). In this work each image is represented by local regions (approximately 30 to 40 local regions per image) that are extracted using the Kadir and Brady detector [Kadir2001]. Each local patch is resized to 48 × 48 pixels, and further divided into four 24 × 24 sub-regions. Each sub-region is then denoted by an 18 dimensional gradient bin similar to SIFT descriptor. Concatenating these four sub-region descriptors together, a 72-dimensional vector for each local patch is obtained. For computational efficiency, each local patch is represented as a 15 dimensional vector by obtaining the first 15 PCA coefficients for each 72 dimensional feature vector.

2.2.3 Part based methods

Probably the most popular technique currently used in object recognition is part-based modeling. Part-based models can capture the essence of most object classes, since they represent both parts' appearance and invariant relations of location and scale between the parts. Part-based models are somewhat resistant to various sources of variability such as within-class variance, partial occlusion and articulation, and they are potentially convenient for indexing in a more complex system [Leibe2004, Lowe2001].

While a part-based model aspires to represent a rigorous geometric relationship among the different parts, it suffers a computational difficulty of having to search among an exponentially large number of hypotheses to solve the correspondence problem [Fergus2005].

A general part-based methodology is described in [Feltzenswalb2005], in the sense that there are no specific appearance or spatial attributes that have to be modeled and the choice is upon the designer. Feltzenswalb and Huttenlocher use pictorial structures introduced in [Fischler1973] but they use a statistical formulation. Pictorial structures model an object as a collection of parts in a deformable configuration represented as spring like connections among pairs of parts. The best match of such a model to an image is found by minimizing an energy function that measures both a match cost for each part and a deformation cost for each pair of connected parts. Appearance and location parameters are learned separately through a simple maximum likelihood estimation procedure. Their work is capable of locating multiple instantiations of an object in an image and the use of dynamic programming reduces the matching complexity significantly.

In [Lazebnik2004], it is proposed to identify groups of local affine regions (image features having a characteristic appearance and elliptical shape) that remain approximately affinely rigid across a range of views of an object, and across multiple instances of the same object class. These groups, termed semi-local affine parts, are learned using correspondence search between pairs of unsegmented and cluttered input images, followed by validation against additional training images.

A study of the degree to which additional spatial constraints improve recognition performance and the tradeoff between representational power and computational complexity is examined in [Crandall2005] through the k-fans model. This model represents both appearance and spatial relationships in a graph structure. The location of the model is simply given by a configuration of its parts where the location of each part is given by a point in the image. The appearance of each part is given by a template based on egdge orientations. Appearance and location parameters of the models are estimated through a classical ML procedure. This work shows that using more powerful models does not necessary improve classification, as it can lead to over-fitting during learning.

In [Fergus2005b], an object is represented in a "star graph" in which the location of the model part is conditioned on the location of a landmark part. In the star model any of the leaf (i.e. non-landmark) parts can be occluded, but the landmark part must always be present. In this approach one of three types of feature types is used: Kadir & Brady [Kadir2001], multi-scale Harris, and Curves. Both region operators give a scale and a location for each feature. A square window around each feature is cropped from the image and rescaled into a k x k patch. Then, its gradient is computed and normalized to remove intensity differences. The normalized gradient patch is subsequently projected into a fixed PCA basis of d dimensions. Curves on the other hand are extracted through a Canny edge detector throughout the image. Edgels are grouped into chains that are broken into their bi-tangent points to form curves. Since the chain may have multiple bi-tangent points, each chain may result in multiple curves (which may overlap in portions). Curves which are very straight tend to be uninformative and are discarded. The curves are then represented in the same way as the regions. Each curve's location is taken as its centroid with the scale being its length. The region around the curve is then cropped from the image and processed in the manner described above. The curve is used as an interest operator, modeling the textured region around the curve, rather than the curve itself. An assignment variable is introduced to assign features to parts. Then joint density is then factored as the product of an appearance term, a relative location term, a relative scale term and a an occlusion term. Expectation maximization is used for learning.

In [Amores2005], the authors define a generalized correlogram descriptor and represent the object as a constellation of such generalized correlograms. Using this representation, both local and contextual information are gathered into the same feature space. They take advantage of this representation in the learning stage, by using a feature selection with boosting that learns both local and contextual information simultaneously and very efficiently. Simultaneously learning both types of information proves to be a faster approach than dealing with them separately.

In [Sudderth2005], features are parts corresponding to SIFT points and their location relative to the object. SIFT descriptors computed on affine covariant regions, model the appearance of the parts. Two types of affine covariant regions are computed for each image. The first is constructed by elliptical shape adaptation about an interest point [Mikolajczyk2002, Schaffalitzky2005]. The second is constructed using the maximally stable procedure of Matas et al [Matas2002]. Both types of regions are represented by ellipses. These are computed at twice the originally detected region size in order for the image appearance to be more discriminating. Each ellipse is mapped to a circle by appropriate scaling along its principal axis and a SIFT descriptor is computed. There is no rotation of the patch. Alternatively, the SIFT descriptor could be computed relative to the dominant gradient orientation within a patch, making the descriptor rotation invariant. Then k-means clustering is used to vector quantize these descriptors, producing a finite dictionary of F appearance patterns. For a more detailed description of the feature extraction process, see [Sivic2005]. Given this feature dictionary, each interest point in an image is described by its position and the best matching descriptor in the dictionary. All feature positions are defined relative to an image-specific coordinate frame, or reference position. Each object category has its own Gaussian prior over reference positions. To generate a feature, a part is independently sampled according to an object specific multinomial distribution over the possible parts. Then, appearance and position are independently sampled. Finally, a Gibbs sampling algorithm in combination with EM is derived for learning the parameters of the model.

In [Opelt2006] the learning algorithm is provided with a set of labeled images where a positive label indicates that a relevant object appears in the image. Objects are not segmented and pose and location are unknown. The image analysis transforms images to grey values and extracts normalized regions (patches) around interest (salient) points to obtain reduced representations of images. Scale invariant, affine invariant and SIFT interest point detectors are examined. As an appropriate representation for the learning procedure, local descriptors of these patches are calculated. Four local descriptors are examined: a vector of all pixels in the patch subsampled by two, intensity moments, SIFTS and moment invariants. Classification is achieved with AdaBoost algorithm. For object recognition, weak hypotheses which indicate if certain feature values appear in images are chosen. For this, a weak hypothesis has to select a feature type τ , its value v, and a similarity threshold θ. Threshold θ is used to decide whether an image contains a feature value v_f that is sufficiently similar to v. The similarity between vf and v is calculated by the Mahalanobis distance for Moment Invariants and by the Euclidean distance for SIFTs. The weak hypotheses finder searches for the optimal weak hypothesis—given labeled representations of the training images ($R(I_1)$, l_1), . . . , ($R(I_m)$, l_m) and their weights w_1,..., w_m calculated by AdaBoost — among all possible feature values and corresponding thresholds. A similar approach is examined in [Opelt2006b].

Vidal et al in [Vidal2003] compare two feature types, generic (wavelets) and informative (fragments), as well as two classification schemes, a simple linear SVM and a more complex Tree-Augmented Network. The results show that with informative features, learning becomes easier and a simple linear separator can reach optimal classification performance. The scope of the paper is to select class-specific image fragments that convey the maximal information about the class. Informative features are selected with a greedy search algorithm from a large pool of parts, typically several tens of thousands, cropped from images containing the class of interest, rectangular in shape, of different sizes and from different locations. The procedure consists of generating a large set of candidate fragments, followed by the computation of the optimal threshold that determines the minimum visual similarity for each fragment to be detected in the image, and finally the selection of informative features. Each fragment X_i is treated as a binary random variable expressing whether it present in the image. This requires a threshold θ_i that represents the minimal detection similarity. The threshold is set automatically by maximizing the mutual information I(X;C) where C is the binary class variably. Thus,

$$\theta_i = \arg\max_{\theta} I(X_i(\theta);C) = \arg\max_{\theta}(H(C) - H(C \mid X_i(\theta)))$$

where H(x) and H(x|y) are Shannon's entropy and conditional entropy. The conditional probabilities $P(X_i(\theta_i)=0|C)$and $P(X_i(\theta_i)=1|C)$ are computed from the training data and the class priors P(C=0) and P(C=1) are chosen a priori. The feature selection process is based on a greedy-search algorithm that iteratively adds fragments to the set of informative features, in a greedy fashion, until adding more fragments no longer increases the estimated information content of the set. In their experiments, each fragment is labeled with the rough location from which it was extracted thus enabling the restriction of the detection zone. This way, fragments obtain a certain degree of translation invariance and capture rough spatial relations among them. Fragments were allowed to move in a 5 x 5 pixel area surrounding their original location.

An extension of the boosting algorithm that is based on GentleBoost [Kohavi1997] is proposed in [Torralba2004]. Learning is performed on large feature vectors that correspond to regions in the image. The proposed algorithm trains a number of binary classifiers jointly instead of training them separately, so that they share as many features as possible. Essentially, at each boosting round various subsets of classes $S \subseteq C$ are examined and a weak classifier that distinguishes that subset from the background is considered. The subset that maximally reduces the error on the weighted training set for all classes is chosen. The best weak learner is then added to the strong learners for all classes $c \in S$ and their weight distributions are updated so as to optimize a multiclass cost function. The method extracts a feature vector of dimension 2000 for each image region of standardized size (32 x 32 pixels). The vector is computed at location x and scale σ by a normalized convolution between the image I_σ at scale σ and filters representing the convolution with a spatial mask. These filters are generated by randomly extracting patches from images that contain objects from the training set, after they are resized to 32 x 32 pixels. A parameter allows the generation of different types of features. For example, in some case, the feature vector encodes the average of the filter responses, which are good for describing textures wheras in another cases, the feature vector becomes better for template matching [Vidal2003]. By changing the spatial mask, they can change the size and location of the region in which the feature is evaluated. This provides a way of generating features that are well localized (good for part based encoding and template matching) and features that provide a global description of the patch (good for texture like objects).

Also, hybrid methods that combine both discriminative and generative machine learning techniques have been recently developed. In [Holub2005], the authors have developed Fisher kernels based on the constellation model. For every input, a Fisher kernel method calculates the Fisher score of the input, and a Support Vector Machine (SVM) is applied for classification in the Fisher score space. In this work, interest points are either manually selected or detected with the Kadir and Brady detector [Kadir2001]. 11 x 11 pixel patches centered at the interest points are extracted. Patches are then scaled to the scale of the detection (a constant scale is assumed in the manual selection of features) and subsampled to 11 x 11 matrices. Appearance parameters are then reduced using PCA. The generative model that is used resembles the one used in [Fergus2005b] and it is used to obtain maximum likelihood estimates of parameter values for each object class. The discriminative model that is used for learning object categories is a weighted subtraction between the ML estimates and the probability of a data-point belonging to any other class, in order to simultaneously pull data-points towards their own class label and push them away from other competing classes.

Boosting appears to be another promising path towards generative and discriminative classification. Bar-Hillel et al. [Hillel2005] have presented a boosting-based method based on their own generative model, which, similar to the constellation model, models part relations as a global distribution function. In this work images are rescaled to have a uniform horizontal length of 200 pixels. The "Kadir and Brady" [Kadir2001] and the "Gao and Vasconcelos" [Gao2004] feature detectors are examined. Both detectors produce an initial set of thousands of salient candidate patches for a typical image. The saliency score of each candidate patch is multiplied by its scale, thus creating a preference for large image patches, which are usually more informative. After their initial detection, selected regions are cropped from the image and scaled down to 11 x 11 pixel patches. The patches are then normalized to have zero mean and variance of 1. After that the patches are represented using their first 15 DCT coefficients (not including the DC). Finally, 3 additional dimensions are concatenated to each feature, corresponding to the x and y image coordinates of the patch, and its scale respectively. Their model contains information about the appearance, scale and location of each feature part. Appearance of different parts is considered independent. Although this is not the case for scale and location, once the object instances are aligned with respect to these parameters the assumption of part location and scale independence becomes reasonable. This is achieved through a three dimensional hidden variable C = (C, Cs). The joint probability of takes the form

$$\mathrm{p}(\{X^k\}_{k=1}^P, C \mid \theta) = p(C \mid \theta)\prod_{k=1}^{p} p(X^k \mid C, \theta^k) = p(C \mid \theta)\prod_{k=1}^{p} p(X_a^k \mid C, \theta_a^k)\, p(X_l^k \mid C_l, C_s, \theta_l^k)\, p(X_s^k \mid C_s, \theta_s^k)$$

where θ is the model parameter and subscripts a, s and l denote the appearance scale and location respectively.

In [Zhang2006], Bayesian classification is combined with a new method for generative part-based object modeling called Random Attributed Relational Graph (RARG) that captures the advantages of the pictorial structure model [Crandall2005] and the constellation model since it accommodates part occlusion and the partial relationship among object parts. Formally a RARG is defined as a quadruple R = (V,E,A,T), where V is the vertex set, E is the edge set, A is a set of random variables that captures the statistics of part appearances and the statistics of part relational features. T is a set of binary random variables, modeling the presence/absence of nodes. The learning process consists of two passes. A generative initialization through variational EM [Zhang2005] is intended to roughly discover the structure of the object model and learn an approximate appearance and spatial relation distribution. The learned generative models are used to find probabilities of the initial part correspondences with which a discriminative learning step is conducted by a new EM procedure. In the experiments, the Kadir and Brady region detector [Kadir2001] is used to define parts. Features extracted from image parts include regular color moments (ten components), size (output from region detector), and spatial coordinates. Relational features include spatial coordinate differences.

2.2.4 Up-to-date panorama of most efficient techniques

In order to have a comparative study of up-to-date most efficient techniques, we synthesized in the following the results of the two most recent evaluation campaigns dedicated to object recognition techniques.

2.2.4.1 PASCAL VOC 2006 (http://www.pascal-network.org/challenges/VOC/)

The PASCAL network of excellence organizes several challenges concerning pattern analysis and statistical learning. One of them, the Visual Object Classes recognition challenge is certainly the most popular benchmark dedicated to object recognition. A description of the 2006 datasets, results and techniques used by the participants is provided in [Everingham2006]. It is notable that among the 20 participants of the classification task, 15 of them used bag of words techniques which indeed give the best results. Other techniques include correspondence based techniques (knn search + vote), classification of individual patches, graph neural network and detection based techniques (Hough transforms or sliding windows). The general scheme of the bag-of-words techniques is the following: local regions extraction + region appearance description + quantisation into visual words + histogram of visual words + class/non-class classifier. The 15 techniques only differ by the choice of these different steps:

- local regions extraction: Most techniques are based on interest points or overlapping grids. Other methods include random position and scale, segmented regions or combinations of previous methods.

- region appearance description: More variety is present for this step although SIFT is the most used one. Other methods include PCA on pixel values, Haar wavelets, gray level moments and invariants, colour histograms, shape context, textures moments and texton histograms and spatial pyramids.

- quantisation into visual words: Single codebooks or multiple codebooks are created mostly by unsupervised clustering (K-means or LGB clustering). Other methods include random cluster centres and supervised clustering.

- histogram of visual words: Most techniques are based on frequency and few of them on presence only.

- class/non-class classifier: Most used and most efficient techniques are based on non linear support vector machines used with a χ^2 kernel. Other methods include linear SVM, regressions or boosting.

The recognition rate of the best techniques range from 86% to 98% depending on the class.

Note that the PASCAL VOC challenge includes a second task consisting in localizing the objects in the images and which is in fact more difficult. The localization rate of best techniques range from 14% to 44% depending on the class. This means that state-of-the-art techniques still make a wide use of contextual information for the classification task. Localizing the object is still a challenging task.

2.2.4.2 Imageval Task 4 2006 (http://www.imageval.org/e_presentation.html)

ImagEVAL is a new image retrieval benchmark initiative that was launched in France in 2006. An interesting aspect that distinguishes ImagEVAL is that its specification and organization were established by both a research team and professional archivists [Moellic2006]. The task definitions were discussed in order to address the real problems that photo agencies face. The images on which the evaluation was conducted were also professional ones. They were selected by professionals, allowing the ground truth to be established in a confident way: as expected by the users, not by the researchers. It is therefore close to real-life scenarios, with challenging image collections and lower performances as a result. The fourth task of the ImagEVAL benchmark was dedicated to object recognition. This database is one of the most challenging available. The objects are in very different poses, contexts and sizes and the classes contain few examples.

Among the 9 runs submitted, the following techniques were used:

- 3 bag-of-words techniques with SVM classifier
- 3 techniques based on global features and SVM classifier
- 1 technique based on graphs of regions matching and SVM classifier
- 1 technique based on local features selection and correspondence based technique (knn search + vote algorithm)
- 1 combination of different techniques (the technique differs from one class to another).

The best run was obtained with the combination of different techniques (MAP=0.22) showing that there is no single technique working better on this kind of objects. Among the other runs (using a single technique), the local features selection technique ranked first (MAP=0.21) and the graphs of regions matching technique ranked 2nd (MAP=0.19). Then come the bag-of-words runs (MAP about 0.17) and finally the runs based on global features (MAP from 0.14 to 0.17). It is notable that contrary to the PASCAL challenge, the bag-of-words techniques do not give the best results, showing that with more realistic and diversified databases, part-based methods taking into account the spatial structure of the images can improve the performance. One possible interpretation is that the higher diversity of the ImagEVAL classes prevents the benefit from contextual information as in PASCAL VOC 2006.

2.2.5 Conclusions

Tremendous progress in generic object recognition has been achieved in the past 5 years, thanks largely to the integration of new data representations, such as invariant semi-local features, developed in the computer vision community with the effective models of data distribution and classification procedures developed in the statistical machine-learning community. Bag-of-words techniques coupled with SVM classifiers have notably become very popular and still give the best results in many evaluations although a lot of effort was spent on more complex part-based models. We thus recommend integrating one technique of the bag-of-words family in VITALAS system as a baseline, with different options for the feature extraction step and the learning stage. However, recent results tend to show that the performances of these techniques are limited when using a more realistic corpus of images such as the one used in VITALAS. Such data involve a higher diversity of the object classes and little contextual redundancy. Recognizing objects in them is still a challenging task that needs further research efforts regarding local shapes representation in part-based models, use of relative geometry between parts and large scale learning strategies.

2.3 Face detection and recognition

VITALAS research is not directly focussed on face detection and recognition problematic since it is the main goal of other research projects. However, a complete multi-modal indexing system must have a face detection and recognition module since the related functionalities are often required by the users. In this section, we present a brief scientific and technological overview in order of face detection and recognition techniques in order to drive the integration of a state-of-the-art face detection and recognition module.

2.3.1 Introduction

Although great efforts in the face detection field were made since the early 70s facial recognition systems became a reality only during the 90s. The first techniques were too rigid and did work only with frontal poses on a plain background [Hjelmas2001].

There are two main approaches about how to deal with more complex face detection issues. The first one is based on the knowledge acquired about the features that a face must have. According to this approach, a face is detected by means of its shape, face's elements or skin colour.

The second approach is based on working with a face as a common pattern recognition problem, treating the face not as a set of features, but as a set of intensities.

According to [Yang2002] the main problems that arise when trying to detect a face are:

- Changes in its position
- Structural elements presence
- Different facial expressions
- Partial face occlusions
- Different environement conditions: illumination, camera type…

2.3.2 Face detection methods

Face detection methods can be classified according to the type of source that holds the face. This means that methods can be classified according to their usefulness in detecting movement or static images, colour or gray scale images or allowing turned face.

Fig. 1 shows a scheme of the face detection methods that are described in the next subchapters.

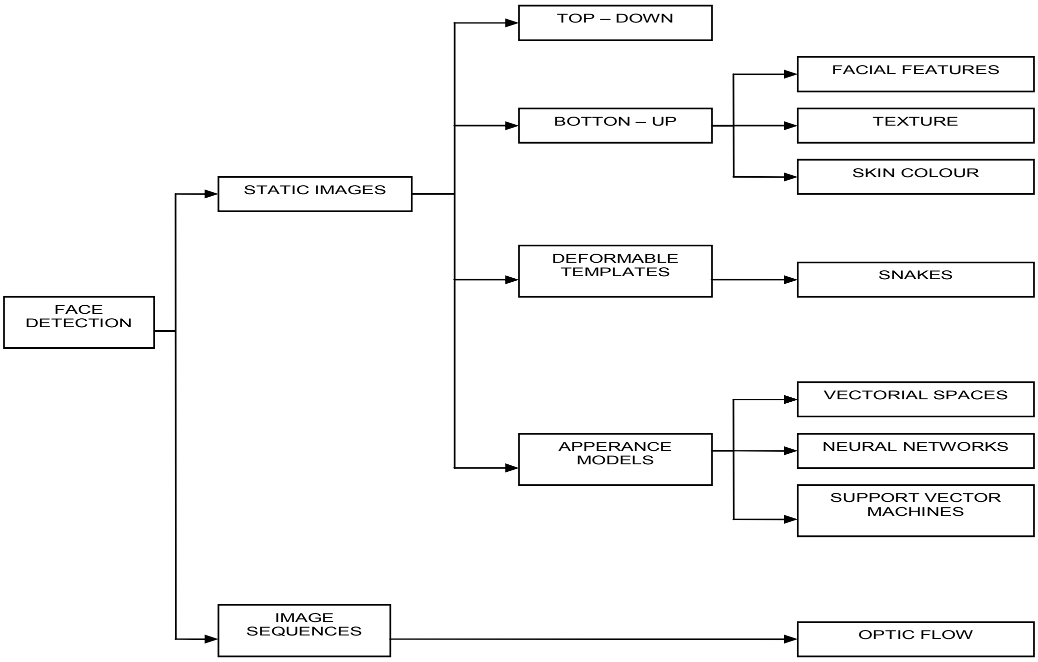


Fig 1. Face detection methods [Hjelmas2001]

2.3.2.1 Face detection in static images

According to [Yang2002], there are different types of methodologies that can be used to deal with the process of detecting faces in static images:

- Knowledge-based methodologies: consist of rules that establish the relationships between the facial features. They encode the human knowledge about what a face is.
- Invariant features-based methodologies: establish some features that an element “face” has when the light, point of view or illumination condition change.
- Patron-based methodologies: compare different patrons of one face with others stored previously.
- Appearance-bases methodologies: Contrarily to the previous methodologies, these ones are based on the appearance of the image, that is, in the intensity pixel values which compound the image.

Next sections analyze the available methods used to detect faces in static images.

2.3.2.1.1 Methods based on top-down knowledge

By means of this approach, the human knowledge is integrated in the facial images. It is expected that two symmetric elements (eyes) were found, usually with two elongated elements in the above part and a nose and mouth too. All of them with a specific position connection among them.

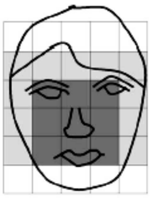


Fig. 2. Different elements that compounds a face [Yang1994]

One of the main problems of this approach is the difficulty that arises when this knowledge has to be transformed in well defined rules. If these rules are too strict, the false negatives number increases. On the contrary, it is the false positives number that increases.

Yang and Huang [Yang1994] use a tree-approach which allows different level rules. In the highest levels (less resolution) the observed features are related to the general appearance of the face. In the lowest levels, the set of rules is linked to the facial features position.

A simple example of the first level rule is proposed by Kotropoulus and Pitas [Kotropoulos1997] and is based on the vertical and horizontal histogram projection and in the detection of the points where the intensity has an abrupt change.

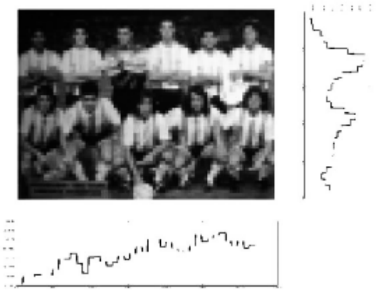


Fig. 3. Histogram projection to the primary selection for facial images

2.3.2.1.2 Methods based on bottom-up knowledge

These methods are based on an opposite approach to the previous methods. Their hypothesis is that there are little invariants that allow humans to recognize the presence of a face in different locations and poses.

According to this approach, this methodology tries to detect a set of facial features and, after that, to infer the existence of the face by means of statistical methods.

2.3.2.1.2.1 Facial features detection

Some authors, as it is mentioned in [Yang2002], have used different methods to locate the different elements that compound a face.

Some of these authors propose searching techniques based on basic machine vision algorithms [Bourel2000][Gu2003]. A very popular method consists on searching for the valleys of the histogram projection in places where the desired feature is placed [Gu2003].

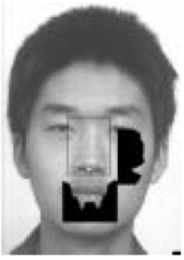


Fig. 4. Localization of facial elements using histogram projection techniques.

Other authors perform the facial features search using patron learning technologies. Heisele [Heisele2001] solves the face elements location problem using the support vector machines methods. This solution consists of the generation of a 3D synthetic face model with multiple renders in several positions. As they are based on a 3D model, Heisele knows a priori the exact location of all the elements of the face that he has to search in the image, so this allows him to generate automatically a wide range of training data.

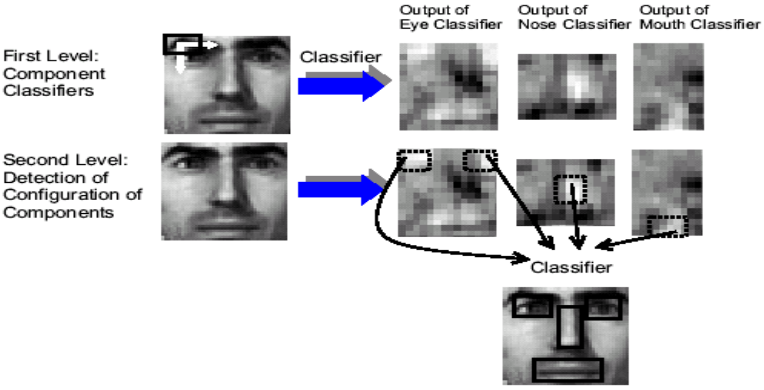

Output of Eye Classifier
Output of Nose Classifier
Output of Mouth Classifier
First Level: Component Classifiers
Classifier
Second Level: Detection of Configuration of Components
Classifier

Fig. 5. Heisele solution to detect facial elements.

Most of these methods [Yang2002] make different checks to verify that the obtained components are correct. Some authors determine the image searching areas from other located elements [Bourel2000] based on anthropometric measures which determine the possible distances where the different facial elements can be located.

Heisele et al. [Heisele2001] use a SVM net (Support Vector Machines net) to test if the located elements position is compatible with the face elements.

2.3.2.1.2.2 Texture

One feature that can determine the presence of a face is the modelization of the associated texture. To get that, some authors [Dai1996] propose the extraction of statistics parameters that define the different textures, including hair, face and others. These parameteres are clustered by a Kohonen neural network (SOM) and then a neural network is used to classify the textures as belonging to a face or not.

Other texture modelization way is based on the creation of statistical models (Gaussians, mixture Gaussians…) that allow to determine the appropriate classification.

2.3.2.1.2.3 Skin colour

It has been studied that although skin colours vary between different persons, this colour has a very small cluster in the colour space, even taking into account the different race tonalities [Hjelmas2001].

Some authors propose the creation of a skin colour model using empiric techniques [Hua2002] based on the YUV colour model. Other authors propose, instead, the generation of a statistical colour model based on examples (Gaussian distributions, mixture Gaussian distributions…)

Some others choose a texture classification in clusters using a Kohonen net (SOM) [Brown2001].

According to [Yang2002], skin colour is not enough to detect or follow a face. It is necessary to combine these methods with a shape, texture or movement analysis. However, it is a good method as primary segmentation that allows to select possible candidates for the analysis.

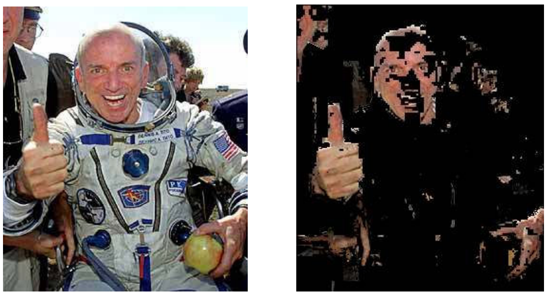


Fig. 6. Skin colour segmentation example

2.3.2.1.3 Deformable templates

Deformable patron models are methods based on using an energy function minimization in order to solve the non-rigid patrons search [Perlibakas2003].

The definition of a deformable patron needs:

- A parametric geometry to model the shape
- An extern energy which models the object's shape restrictions (curvature, size…)
- An extern energy which models how the object is watched in the image (borders, colour…)

So the template energy is defined with the following equation:

$$E_{Template} = K_1 \cdot E_{\text{int}\,erna} + K_2 \cdot E_{externa}$$

This energy depends on the template parameters, so that the more minimized it is, the more likeness has the patron with the object.

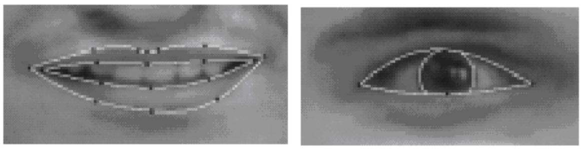


Fig. 7. Deformable template example [Malciu2000]

A particularization of deformable patrons are the Snakes [Gunn1998]. They consist of a set of deformable patrons defined by a set of points P(x,y) that compound a polyline or a spline.

One disadvantage of this methodology is that these deformable contours must be initialized in the nearly real contour zones. So it is useful to combine the Snakes method with other approaches that allow to detect the candidate elements.

2.3.2.1.4 Appearance based methods

Previous methods are based on finding templates that have been defined by "experts". The appearance-based detection approach is more oriented to the automatic patron learning from examples. These methods try to search the differentiating features between face images and non-face images.

There exist different ways to carry out this differentiation, such as:

- To calculate the Bayesian probability of belonging or not to the "face" class or not
- To project each "face" element on a vectorial space with less dimensions where the differentiation can be processed (PCA - Principal Component Analysis)
- To map an acceptance region by means of neural networks
- To project the "face" element on a vectorial space with more dimensions where the classification can be processed (SVM – Support Vector Machine)

2.3.2.1.4.1 Systems based on linear vectorial spaces

Principal Components Analysis

The methods based on PCA [Hjelmas2001] project an image on a vectorial space with less dimensions obtained by means of statistical techniques, such that the maximum variability is conserved.

To detect faces in an image, the difference between the original image and the projected one in the transformed space and its reconstruction [Hjelmas2001] are analyzed. If the reconstructed image doesn't belong to the "face" class, its reconstruction will be very different to the original image.

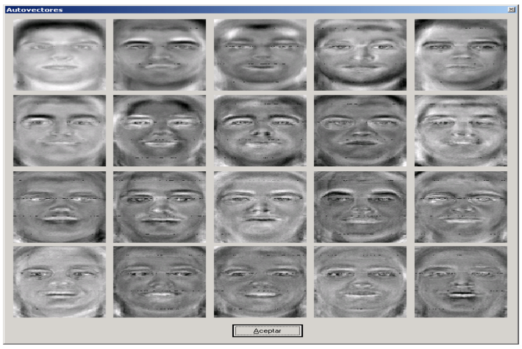


Fig. 8. EigenFaces example

The use of PCA is an intuitive and appropriate way to make a subspace that represents a class, but in the case of the faces, it is not necessarily optimal [Hjelmas2001].

Because of that, Sung and Poggio [Sung1998] suggested a method to generate a Gaussian cluster set that defines the face subspace. This subspace couldn't have been modelled by means of PCA (fig. 9).

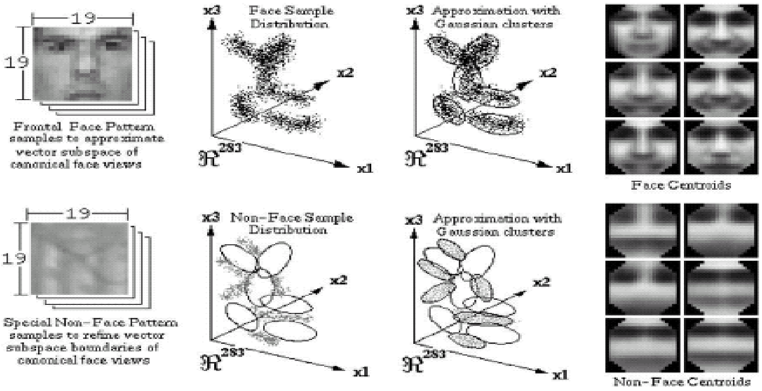

19
19
Frontal Face Pattern samples to approximate vector subspace of canonical face views
x3 Face Sample Distribution
x2
x1
ℜ283
x3 Approximation with Gaussian clusters
x2
x1
ℜ283
Face Centroids
19
19
Special Non-Face Pattern samples to refine vector subspace boundaries of canonical face views
x3 Non-Face Sample Distribution
x2
x1
ℜ283
x3 Approximation with Gaussian clusters
x2
x1
ℜ283
Non-Face Centroids

Fig. 9. Face images vectorial subspace

Sung and Poggio generated 12 clusters, 6 with faces and 6 with others elements. Using the mean and the covariance matrix of each cluster, they check the Euclidean and the Mahalanobis distance with every cluster. These distances, introduced in a neural network, can classify the image as belonging to the “face” class or not.

2.3.2.1.4.2 Neural networks

Neural networks have become an appropriate method to solve patron recognition problems. The developed tests are not based only on the use of multilayer perceptrons, since there are different detection architectures and neural networks combinations anywhere in the detection process area.

In this way, Rowley [Rowley1998] suggests a system which uses a set of neural networks to analyze the possible face image, dividing the image into pieces. He also suggests a neural network that can estimate the face angle in order to detect turned faces.

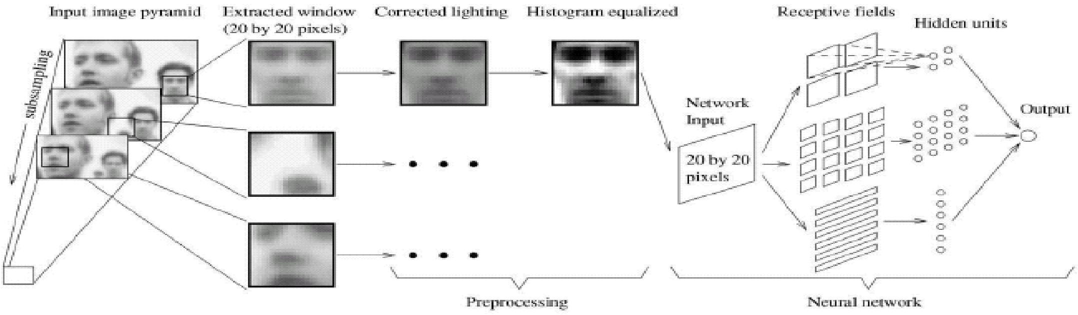


Fig. 10 Rowley’s suggested architecture

Another architecture based on neural networks has been suggested by Yang et Al [Yang1999]. This architecture is based on dispersed networks that use the *Winnow rule* in the training step.This system is recommended in cases of numerous features where not all of them converge to the solution.

2.3.2.1.4.3 Support vector machines

Another training way for statistical, PCA or neural network system, consist of the use of SVM (Support Vector Machines). This technique allows projecting the features of a class on a vectorial space with more dimensions, allowing greater dispersion and easier separation. Osuna et al. [Osuna1997] were the pioneers that applied this method to facial detection, and they got great advances in the detection speed.

2.3.2.2 Face detection in video images sequences

Image detection in video sequences is, a priori, easier than detection in static images [Yang2004a], because in this analysis the additional information provided by movement can be used.

By means of the image optic flow it is possible to distinguish the background movement from the movement of the object and its features. In this way, a basic treatment of the optic flow allows a significant reduction of the face searching area.

There are studies [Gunn1998] that use the optic flow as a constriction equation in order to solve the 3D deformable parameters model of the face.

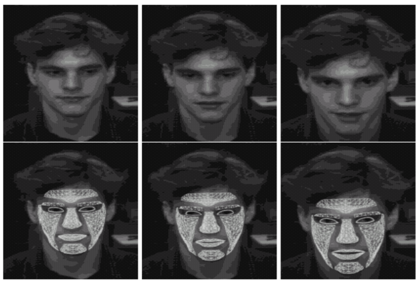


Fig. 11. 3D face model example [Gunn1998]

2.3.3 Face recognition applications

In the following subchapters, a set of different face recognition systems is shown. In these systems, a wide range of approaches to the face recognition algorithms are represented, such as 3D models, 2D models, static face image analysis, movement face image analysis…

2.3.3.1 A4Vision

The A4Vision [A4Vision2007] product line consists of a combination of Hardware and Software built around the concept of 3D facial reconstruction and recognition. The hardware devices and applications have been designed for use in the Physical Access Control, Time and Attendance, and Civil Identification markets.

Fig 12 (a) A4Vision reader (b) A4Vision Camera

The *Vision Access – 3D Face Reader* (Fig. 1a) was designed for quick and easy user positioning which optimizes throughput at entrances. The system is designed both as a stand alone system or to interface with legacy access systems in a distributed networked environment. Through a proprietary matching engine and algorithms, this system performs subject identification and verification in less than one second. The Vision Access face reader is comprised of a real-time 3D surface scanner working in invisible near-infrared light and can be used in both identification and verification modes.

The *Vision 3D + 2D ICAO camera* (Fig 1b) is used to perform enrolment, verification and identification of 3D and 2D face images. The camera simultaneously captures 3D and 2D images that comply with ISO/IEC CD 19794-5 (Standard for ICAO compliant images for machine readable documents). The device is designed with A4Vision's advanced optical technology, structured light, and algorithms, using a special projector and a digital camera. The system performs this processing and searching in real time.

Both devices need a previous enrolment of the person in the reader to generate the 3D biometrical profile, and it is also necessary to use these equipments to perform the following readings. Therefore, it is impossible to apply this technology to a 2D information source such as pictures or videos.

2.3.3.2 Cognitec

Cognitec Systems [Cognitec2007] develops and markets the FaceVACS® face recognition software. The installation of the SmartGate system at Sydney International Airport in 2002 was a major step towards a solution for automated border control using Cognitec's software.

Cognitec has different products according to the possible use and implementation. For example, with *FaceVACS – Acquisition* it is possible to capture digital portraits for ID documents. It can provide a graphical user interface to visually assess the checked portrait characteristics like frontal pose, uniform lighting, glasses, eyes open and geometrical requirements.

Using *FaceVACS®-Entry* it is possible to add face recognition to conventional access control systems. At the access point, the face of every person is captured by a video camera and the facial features are extracted and compared with the stored features. In addition, visual control by security personnel is supported.

The *FaceVACS-DBScan* is used to support biometric identification, comparison, similarity check and verification of persons that are members of a large group. Data consist of databases of biometric identities and closely related data. The biometric identities are trusted face photographs.

The *FaceVACS-Alert* system automatically scans incoming video streams, detects multiple faces and checks if the found faces match a watch list of persons; in case of a match, operators are notified - all in real time. Applications include the identification of unwanted people at airports or railway stations, in sports stadiums, shopping malls or schools, but also the identification of high ranking customers in order to offer them special services.

Although these systems can provide good features in order to detect users' faces, results are linked to the controlled environment conditions. So in order to detect and recognize the different users, it is necessary to have a known background with a controlled illumination to achieve the optimum conditions.

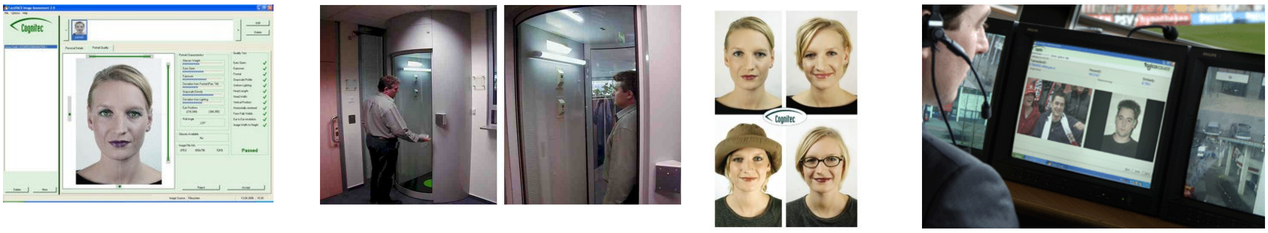


Fig 13: a) FaceVACS – Acquisition b) FaceVACS®-Entry c) FaceVACS-DBScan d) FaceVACS-Alert

2.3.3.3 FaceSnap Recorder

FaceSnap® RECORDER [FaceSnap2007] is a turnkey solution for video surveillance, monitoring, law enforcement and other applications requiring face recognition and recording. A combined digital video recorder and face recognition software locates and extracts facial images from video footage for identification and/or verification. Individual faces and their facial landmarks are then recorded and stored in an easy-to-navigate database that can be viewed on a PC monitor.

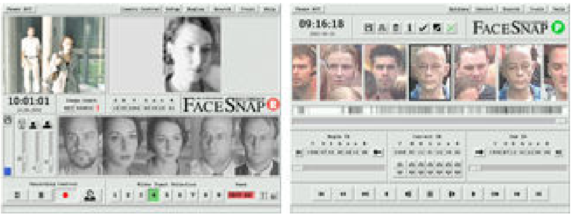


Fig. 17: FaceSnap Recorder

The system recognizes and records the same face from several angles as well as simultaneously switching between multiple faces entering its field of vision. It's quick, efficient and reliable.

The graphic interface enables any operator to use all functions with just a minimum training. For maximum user control, the database can be searched manually or with an automatic search function.

This system is a stand-alone system, featuring a digital video recorder and all necessary software. The recorder has video inputs from a variety of formats and can also operate in a network environment.

However, the system is developed to work with a fixed camera and zoom. It also requires a previous knowledge of the image background to optimize the detection features.

2.3.3.4 FaceIt Argus

FaceIt Argus [FaceIt2007] is a real-time facial screening system that captures faces in live video stream from high resolution cameras, searches them against predefined watch lists, and generates alarms whenever a match is found.

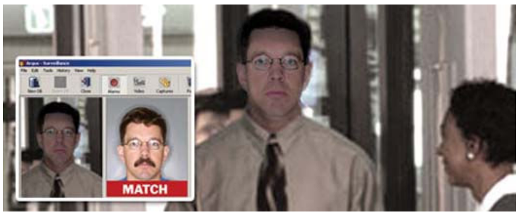


Fig. 18: FaceIt Argus

The system can use a multi-View enroller to compensate pose and angle variations. Its matching technology combines facial geometry and skin texture for maximum accuracy and it is possible to adjust thresholds to accommodate to the changing security needs.

But, as happens with the previous systems, FaceIt Argus was developed to work with a fixed camera and cannot work properly if the background suffers continuous changes.

2.3.3.5 VeriLook Face Identification

Neurotechnologija has developed a PC-based face recognition algorithm (VeriLook) [VeriLook2007] designed for biometrical system integrators. It offers capabilities of the most advanced and convenient facial identification systems at a reasonable cost.

Its face recognition algorithm implements advanced location of faces, enrolment and matching using robust digital image processing algorithms. It is fast and accurate and allows the location of multiple faces in live video streams and still images. Its features generalization mode generates the collection of the generalized face features from several images of the same subject. Then, each face image is processed, features are extracted, and the collections of features are analyzed and combined into a single generalized features collection, which is written to the database.

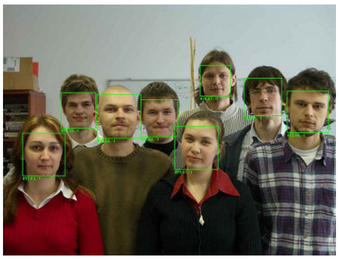


Fig. 19: VeriLook Face Identification

This system can provide good results and has a good processing time, but it requires high resolution images (more than 640x480 pixels) and not always is possible to have this minimum image quality.

2.3.4 Conclusions

According to the previous overview of face recognition existing systems, it seems that current applications have a set of common constraints:

- A fixed environment. Systems have a fixed camera surveying the environment. Not only the background but also the expected face location in the image is known. Furthermore, some systems require their own equipment to work properly.
- A fixed resolution and size. The analyzed systems always work with a fixed camera resolution and image size and it is always the same for each system.
- Neither of them supports zoom in the processed image or videos.

On the other side, VITALAS project deals with very heterogeneous contents coming from different generalist datasets. Ideally, face recognition should thus work in a non controlled environment, with variable image features, various face poses and various scales. Clearly, such a hard task is still an open problem and will not be solved within VITALAS project since it is not its main focus.

However, the non-controlled environment issue can be partially addressed under certain conditions such as frontal views and a limited number of targeted people (with enough learning materials). For the following of VITALAS, we thus recommend to first focus on the development of an efficient background independent face detection algorithm. An existing common recognition technique, such as PCA, could then be integrated to perform recognition tests on a limited number of people for which we have a sufficient number of frontal views.

3 Audio Content

The goal of extracting the mono media audio features introduced in this section is to provide a *rich transcription* of an audio document [Zweig2006], i.e. a transcription that contains both structural information such as segment boundaries, segment types or identified programme jingles as well as information about the actual spoken words. Such extended feature sets can be used to describe and classify an audio document at a higher level of abstraction.

Within this document, an audio document is considered as a sequence of audio samples without any additional meta information, i.e. the audio document is assumed to be completely *unstructured*. As this document presents state of the art approaches regarding mono-media indexing, only audio is considered here, even if it is derived from the audio track of a video file.

The first section introduces common low level features which can be extracted from audio data directly. These can be applied for extracting the higher level features described in the following sections. Some low level features - such as energy or zero crossing – can be used directly to extract a higher level feature by applying simple Gaussian mixture classifiers. Other high level features – such as the spoken language features - require more complex classification processes.

Figure 2 illustrates a common workflow for audio content extraction. Based on the given audio data, more and more complex features are derived in a sequential classification process. The possible output of the content extraction ranges from structural features to multi level speech transcripts. Besides the input from the preceding steps, each extraction method requires various low level features derived directly from the audio data as described in the subsections below.

Some principle concepts have already been introduced in deliverable D3.1.1 “State of the Art in Cross-Media” in the section “Audio Annotation via Supervised Learning techniques”.

3.1 Low level audio features

Feature sets for discrimination between sounds are typically based on the spectral characteristics of the audio signal. The common representatives of this feature class are introduced. Depending on the classification task, these low level feature sets are augmented with additional characteristic low level features such as energy or zero crossing, which are presented at the end of the section. Adequate dimensionality reduction techniques are described, which are crucial in the scalability context of VITALAS.

3.1.1 Spectral features

Feature vectors based on the spectrogram of the audio signal are an important input for most audio analysis tasks such as segmentation, speaker clustering or speech recognition. Standard coefficients from signal processing, such as Linear Prediction Cepstral Coefficients (LPCC) [Huang2001], Mel Frequency Cepstral Coefficients (MFCC) [Davis1980] or Perceptual Linear Prediction (PLP) [Hermansky1991] are applied to derive the spectral feature vectors.

In [Beyerlein2002], no consistent improvement was observed when using the PLP coefficients or when combining MFCCs with psychophysical concepts as described in [Woodland1998].

The spectral vectors are calculated from a short time window of about 20-30ms. To reduce the amount of data, the windows are applied at an interval of typically 10ms (Humans can distinguish acoustic events with a precision of about 2ms).

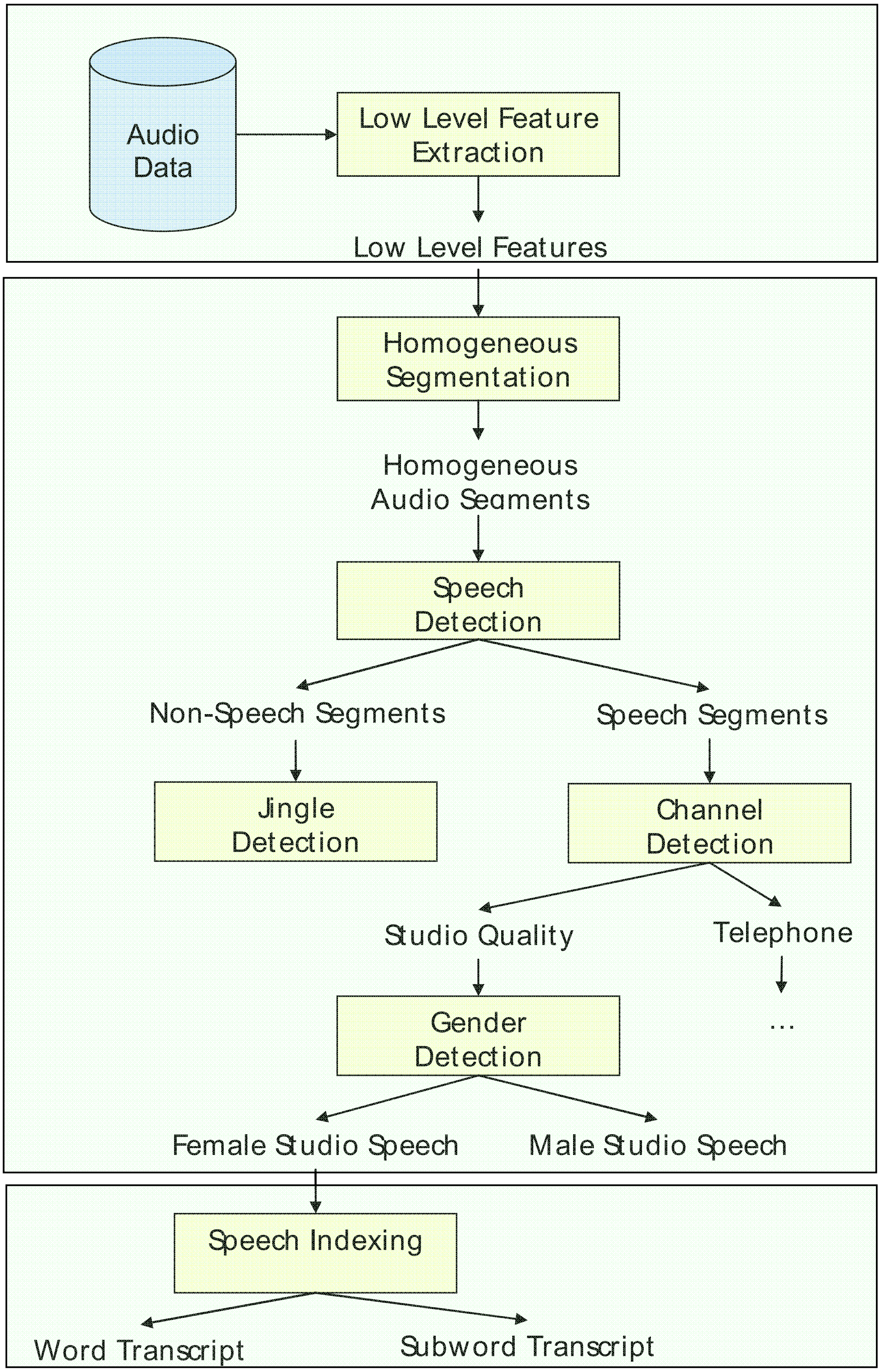


Figure 2: Typical Audio Content Extraction Workflow

3.1.2 Energy

The energy value – sometimes referred to as *volume* [Wang2000] - is often added to larger feature sets. Moreover it can be used on its own as an indicator for signal activity and can provide an estimation for silence detection.

For a set of *n* samples, the energy *E* is typically defined as the log of the total signal energy observed in all *n* samples:

$E = \log\sum_{i=1}^{n}\left(s_i^2\right)$, where s_i is the signal energy of sample *i*.

3.1.3 Zero crossing

A zero crossing in a set of n integer samples is a sign change between two adjacent samples. If a single speech source can be assumed, the zero crossing rate is a simple indicator for the pitch of the signal and can be used for gender detection (see section on structural features). Moreover, it is an effective and robust measure for detecting unvoiced speech, which typically has a low energy but a high zero crossing rate [Wang2000].

The zero crossing rate Z of a set of n samples is:

$$Z = \frac{1}{n} \sum_{i=1,\ldots,n-1 \text{ where } \left(s_i s_{i+1} < 0\right)} 1$$

where s_i is the signal energy of sample i.

3.1.4 Temporal modelling using low level features

The feature vector can be augmented by adding first and second order derivatives of the original feature vector components, modelling short time temporal changes. In [Gales2006], even third order derivatives are integrated into the vectors. The drawback of integrating the temporal information is the linear increase of the vector dimension which is addressed by dimensionality reduction.

3.1.5 Low level feature combination and dimensionality reduction

The low level features can be combined and used in the structure and indexing classifiers described below. Typical feature combinations are introduced in the respective sections.

Standard techniques for dimensionality reduction – such as Principal Component Analysis (PCA) or Linear Discriminant Analysis (LDA) - are often applied to optimize the feature space [Eisele1996]. In [Gales2006], 13 PLP coefficients are extracted and augmented with first, second and third order derivatives, yielding a 52-dimensional feature vector. Then heteroscedastic linear discriminant analysis (HLDA) [Saon2000] is applied to reduce the dimensionality to 39 vector components.

3.2 Structural features

The first step in audio content extraction after extracting the low level features is the segmentation of the audio signal (see Figure 2). After the boundaries between homogeneous segments have been extracted, each segment can be labelled using various classifiers, e.g. for detecting speech or gender. As an additional source of information, programmes can be identified and located using the corresponding programme jingle.

Segmentation and segment classification are required for several reasons:

Richness of the transcription. As reported by INA in the Vitalas D1.1 deliverable on use cases and user requirements, the pre-segmentation of unstructured video data is an important use case. The information can be used by the archivists to speed up the documentation process. Additional information extracted by the jingle detection can serve as an identifier for important programmes that have to be documented manually. Structural features are particularly important for classifying segments that contain a large number of speech recognition errors [Ohtsuki2006].

Indexing scalability. The time-consuming transcription algorithms only have to be applied to segments that contain pure speech, while the speech indexing process can discard all non-speech segments in advance. Internal experiments carried out at Fraunhofer IAIS using German broadcast data showed that only about 25% of the audio data contained pure speech. The indexing of a single speech segment can be carried out independently of the other segments, hence the identified speech segments can be sent to different CPUs for parallel processing. This is particularly useful when analysing large scale audio archives as those covered by VITALAS.

Indexing accuracy. In the context of VITALAS, highly heterogeneous audio material from the broadcast domain has to be indexed. Additional metadata about a segment – i.e. the gender of the speaker or the used transmission channel – can be used to select a dedicated acoustic model optimized for the automatically detected situation.

As illustrated in Figure 2, the extraction of structural features is a sequential process, starting with a segmentation into homogenous audio segments. The extracted segments are then classified at various levels of abstraction.

3.2.1 Homogenous segmentation

Several state of the art systems for broadcast news indexing such as [Gales2006] base the initial segmentation of the audio data on the Bayesian Information Criterion (BIC) as initially proposed in [Chen1998]. The approach detects acoustic changes by estimating the optimal number of probabilistic models for a given segment. An segment boundary is detected if two different probabilistic models explain the acoustic properties better than a single model.

3.2.2 Segment classification

The list of segment classifiers we present in this section contains the most important approaches that have been successfully applied in the context of broadcast data indexing. Based on the classification of a speech segment, an appropriate acoustic model can be selected for speech indexing [Gales2006, Matsoukas2006].

3.2.2.1 Speech detection

In [Biatov2006], Gaussian Mixture Models (GMMs) are successfully applied to detect whether a segment contains speech or not. During low level feature extraction, the frontend calculates the mel cepstrum frequency coefficients (MFCCs) and adds the signal energy to the feature vector. The vector is extended using deltas and delta-deltas of both MFCCs and energy.

In a two step process, each feature vector of a segment is first classified on its own whether it is supposed to be speech or not. Two GMMs with 1024 mixtures are used for speech and non-speech modelling, respectively. The GMMs are trained on 3 hours of hand labelled acoustic data recorded from German broadcast news programmes. The resulting frame level segmentation separates clean high quality speech from segment types such as music, speech with music, singing, telephone speech, auditorium speech or noise. In the second step, another classifier decides whether the segment should be labelled as speech or not, based on the frame level decisions.

3.2.2.2 Channel detection

The different signal characteristics of wide-band studio quality and narrow-band telephone speech require different acoustic models for speech recognition (see below), hence it is important to detect the transmission channel that was used to record the audio data of a certain segment. GMMs can be applied for a simple yet effective bandwidth detection [Barras2006], using the same frontend setup as described above for speech detection.

3.2.2.3 Gender detection

Both [Barras2006] and [Biatov2006] suggest using GMMs for gender detection. The latter suggests to use a feature vector consisting only of the first 12 MFCCs without energy and temporal information.

3.2.3 Jingle detection

To identify important programmes, it is possible to detect the location of the jingle corresponding to a certain programme. A typical jingle detection workflow as used in [Johnson2000] is depicted by

Figure 3.

In a pre-processing step, a signature is extracted from each jingle that should be located in the audio archive. Such a signature can be based on the low level features described in the previous sections. All jingle signatures are then stored in a jingle database for later pattern matching with the archived data. The same feature types used for creating the signatures are extracted from each archived data file. A windowing function is applied to the feature sequence, yielding a set of feature vectors from which a signature can be derived for each window position. Finally, a distance measure is applied to detect the probability that the given jingle is located at the current window position.

Several approaches that have been successfully applied use such a workflow for jingle detection, but some important differences can be observed:

- The type of the underlying low level feature. In [Pinquier2004], energy-normalized spectral coefficients are extracted. Perceptual linear prediction is applied to extract the features in [Johnson2000]. Internal experiments at Fraunhofer showed that standard mel frequency cepstrum coefficients as applied in speech recognition can be successfully used for modelling the jingle signatures.

- The distance measure for comparing jingle signatures. In [Pinquier2004], the Euclidean distance is used for assessing the similarity between two jingles. In [Johnson2000], Arithmetic Harmonic Sphericity (AHS) is applied to determine the similarity between two different signatures. First, the covariance matrix $\sum_X$ of the jingle signature *X* has to be calculated from all feature vectors extracted from the jingle audio data. Then the covariance matrix $\sum_Y$ of the current archive window *Y* is calculated. In a *D*-dimensional feature space, the AHS estimate becomes

  $$d(x,y) = \log\left[tr\left(\Sigma_y \Sigma_x^{-1}\right) \cdot tr\left(\Sigma_x \Sigma_y^{-1}\right)\right] - 2\log\left(D\right)$$

  where *tr(A)* is the *trace* of the *D*-dimensional matrix *A*. Experiments presented in [Johnson2000] show that the AHS distance measure outperforms the Euclidean distance with respect to detection accuracy. On the other hand, AHS is computationally more expensive compared to the simpler Euclidean distance.

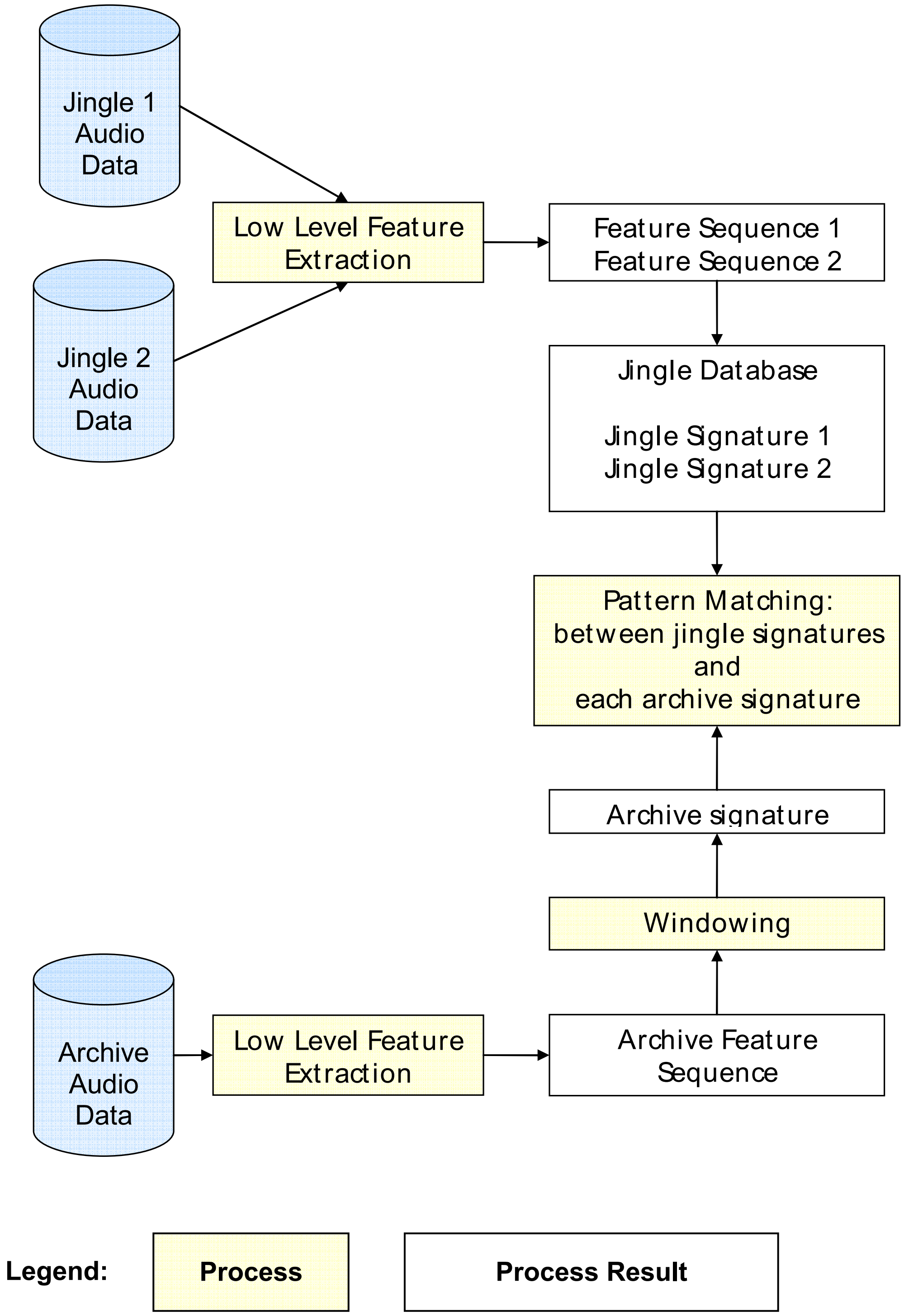


Figure 3: Jingle Detection Workflow

3.3 Speech indexing

The term *speech indexing* refers to the task of extracting full or partial transcriptions of speech contained in an audio document. There are several levels of detail at which the *transcription* of a spoken document could be generated:

- Keyword spotting. In [Wilpon90], an approach is proposed for identifying keywords in unconstrained speech, i.e. without restricting the speakers in any way prior to the recording the data. Phoneme-based garbage modelling is applied for matching the non-keyword parts of the audio signal using a grammar as depicted by Figure 4. Experiments at Fraunhofer IAIS showed that the approach is only effective for a small set of keywords [Schneider07]. Moreover, the set of keywords has to be known prior to the actual recognition process, i.e. all retrieval requests have to be anticipated by the keyword list designer.

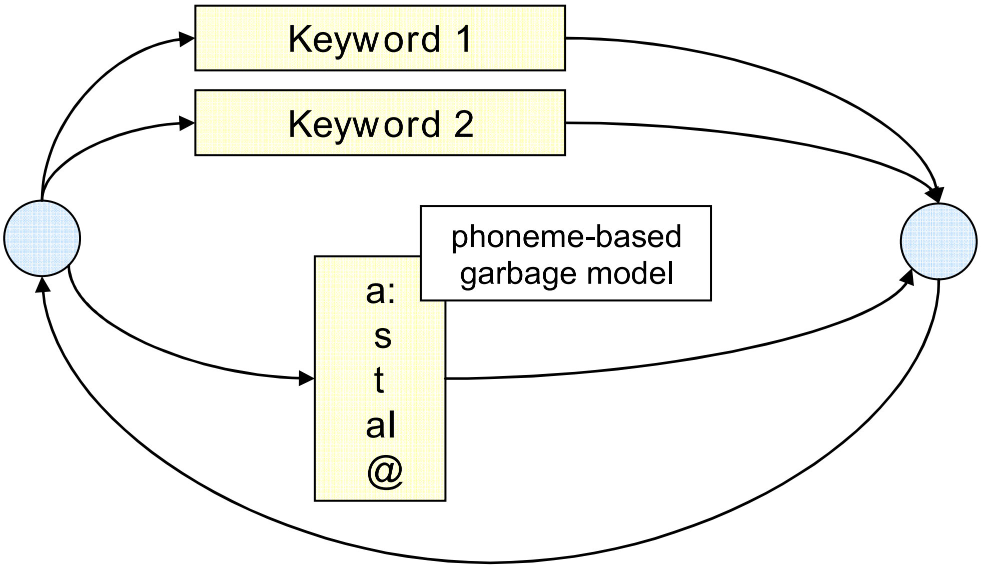


Figure 4: Grammar for Keyword Spotting Using Garbage Modelling

- Full text transcription. This is the most desirable but most difficult task. Several sophisticated state of the art systems such as [Gales2006, Chen2006a, Stolcke2006, Matsoukas2006] try to estimate a word level transcription of broadcast news data, but even the most successful approaches have to cope with word error rates of about 10-20%. The performance degrades even more if more complex acoustic environments are taken into account (see section on acoustic modelling).

  One of the main drawbacks of full text transcription estimation is the fixed content of the vocabulary. The lexicon – i.e. the words that can be recognized – has to be known in advance, and even with state of the art recognizers, its size is limited. For example, in [Gales2006], an error rate of about 10% is reached on a fixed 60.000 word vocabulary. This does not present a problem in a simple dictation application, where the vocabulary can be adjusted to a certain situation (such as the medical or judicial domain), but in broadcast news, the topic of the programme is not known in advance, i.e. before the recognition.

Relevant search terms might not be of interest at time of transcription. They will not be part of the lexicon and can thus never be recognized.

- Sub-word transcription. Approaches proposed in [Ng1998], [Wechsler1998] or [Larson2003] estimate the transcription on a sub-word level, i.e. a phoneme or syllable transcription. Unlike full text transcription systems, such sub-word approaches do not face the Out-Of-Vocabulary challenge. The fixed sub-word vocabulary does not present a problem as the phoneme or syllable set is completely known in advance. Word level keywords can be retrieved later using their sub-word equivalent, which can be derived automatically using grapheme-to-phoneme conversion [Klabbers2001].

Due to the mentioned limitation regarding concept detection scalability, classical keyword spotting approaches will not be considered in more detail. The descriptions below will introduce the concepts of speech recognition for full text transcriptions, and state of the art techniques for sub-word indexing will be presented. Both approaches use the same underlying concepts – phonetic, acoustic and linguistic knowledge resources – but differ in detail.

3.3.1 Typical system setup

The holistic statistical approach to large vocabulary speech recognition has become a standard in both research and commercial systems. Several state of the art systems in the field of broadcast news indexing such as [Gales2006, Chen2006a, Stolcke2006, Matsoukas2006] share the same typical system setup that will be introduced in this section.

In a statistical speech recognition system as illustrated by Figure 5, the goal is to find the word sequence that best matches the observed speech audio data. For this goal, several components are integrated.

- The feature extraction frontend: Low level features such as MFCCs or LPCCs are extracted from the observed speech samples. Dimensionality reduction techniques are applied to optimize the feature space.

- The probabilistic knowledge sources: The pronunciation lexicon, the acoustic and the language model provide a statistical model of human speech.

- Search: The search process identifies the model that provides the best explanation for the observed speech data.

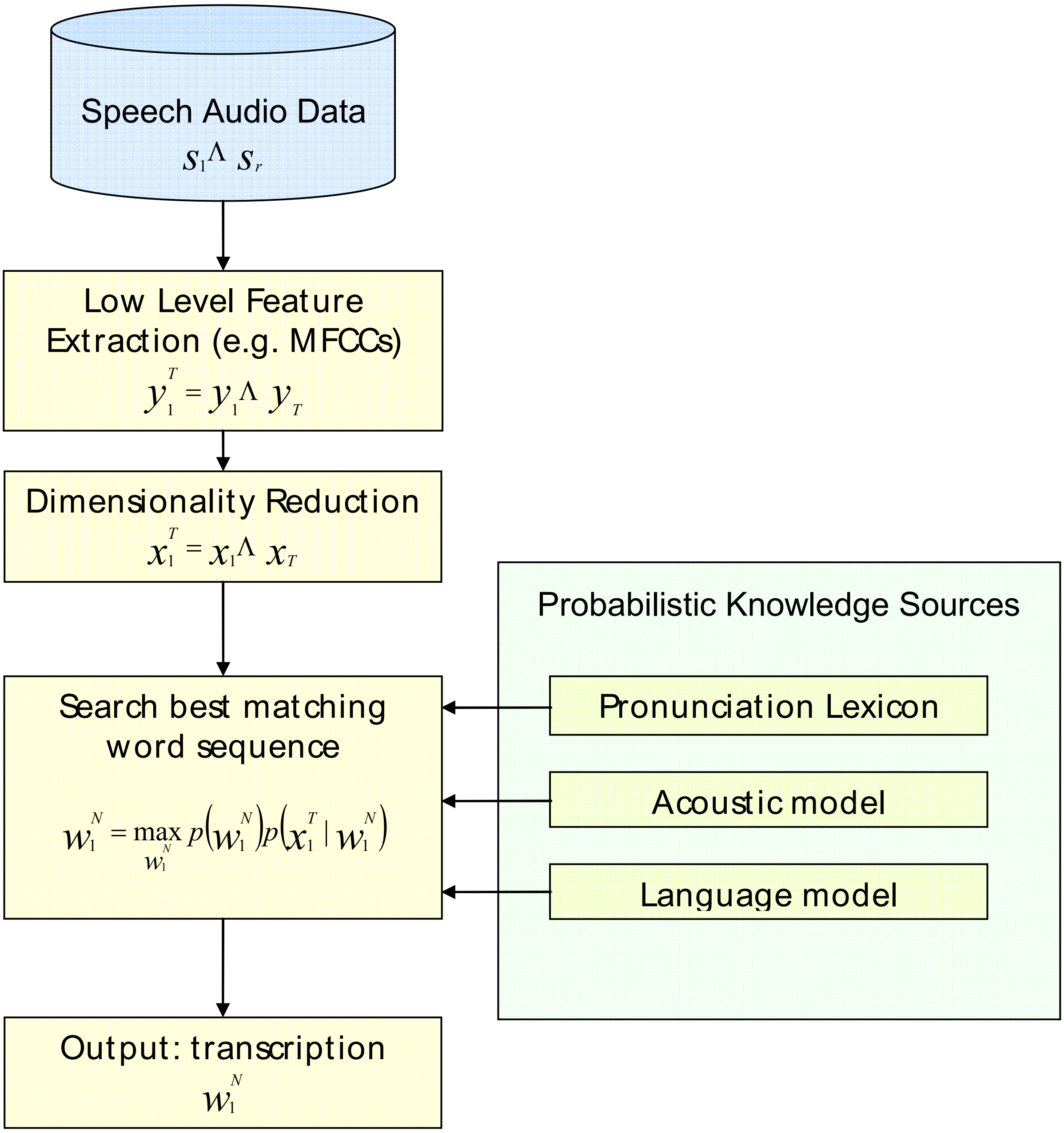

Speech Audio Data
$s_1 \Lambda \, s_r$
Low Level Feature Extraction (e.g. MFCCs)
$y_1^T = y_1 \Lambda \, y_T$
Dimensionality Reduction
$x_1^T = x_1 \Lambda \, x_T$
Probabilistic Knowledge Sources
Search best matching word sequence
$w_1^N = \max_{w_1^N} p(w_1^N) p(x_1^T \mid w_1^N)$
Pronunciation Lexicon
Acoustic model
Language model
Output: transcription
w_1^N

Figure 5: Speech Recognition System Setup

3.3.2 Vocabulary size and search space Complexity

One of the main challenges when approaching automatic speech indexing is the high dimensional search space, where the size of the vocabulary plays an important role regarding both efficiency and accuracy.

Considering a large vocabulary task with a 100.000 word vocabulary and an average sentence length of 10 words, the number of different sentences becomes $(100.000)^{10}$. Clearly, not all possibilities can be taken into account. Reducing the search space can result in wrong decisions at an early stage during search, leading to unrecoverable errors. Effective methods for pruning the search space by removing unlikely hypotheses as described in [Ney1997] are required for efficient decoding.

3.3.3 Phonetic modelling

Using Huang's definition in [Huang2001], a *phoneme* is a unit "of speech sound […] that can serve to distinguish one word from another". Phonemes are the smallest units than can be used for acoustic modelling on a sub-word level. The actual acoustic realization of a phoneme is denoted by the term *phone*.

Compared to the set of possible words or syllables, the set of existing phonemes is rather small, e.g. only about 50 different phonemes can be used for modelling the words of the German language.

For a large vocabulary continuous speech recognition (LVCSR) system, the sub-word form of each word that should be recognized must be known in advance, i.e. a pronunciation lexicon containing the *phonetization* of each word in the vocabulary must be designed.

Typically, automatic phonetization tools as known from text-to-speech systems such as BOSS [Klabbers2001] are used to derive an initial sub-word form of the lexicon. The automatic suggestions should be manually corrected, as the quality of the pronunciation lexicon plays an important role for recognition accuracy [Lamel1996]. In [Mctait2003], it is suggested to correct only the most frequent words manually, yielding an absolute WER improvement of up to 2.6%.

3.3.4 Acoustic modelling

The already mentioned state of the art speech indexing systems [Gales2006, Chen2006a, Stolcke2006, Matsoukas2006] apply statistical acoustic modelling to specify how the phonemes are typically pronounced. The main reason for pursuing a – potentially error prone - statistical approach is the high variability of the input data. The acoustic signal corresponding to the same phoneme varies greatly between speakers due to different voices, accents or speaking rates. Even if the same word is pronounced twice by the same speaker, highly different corresponding sequences of feature vectors can be observed.

Another source of variability is the transmission channel, i.e. recordings from a mobile telephone have significantly different signal properties than high quality studio recordings. As already mentioned in the section on structural features, the knowledge derived from segmentation and segment classification can be used to partly cope with the variability challenge.

The remaining variability must be handled by the statistical acoustic models, specifying the acoustic realization of the phonemes that should be recognized. The models are trained on a large amount of acoustic training data recorded by a wide range of different speakers. Hidden Markov Models (HMMs) have become a de facto standard for statistical modelling of phonemes. Each state of the

HMM corresponds to a part of the phoneme that should be modelled and emits a certain acoustic output with an emission probability derived from a statistical training process.

Gaussian or Laplacian mixture densities are used to model the emission probabilities. The HMM transition probabilities model the speaking rate variability such that a standard speaking rate is assigned a high probability. The transition probabilities and the parameters of the probability density functions are estimated during the training phase. Computationally expensive discriminative training criteria such as Maximum Mutual Information Estimation (MMIE) have become feasible in the last years [Woodland2000], yielding significant improvements compared to the standard maximum likelihood approach based on Expectation-Maximization [Bilmes1998]. Using transition and emission probabilities, the probability that a certain speech utterance corresponds to a certain Hidden Markov phoneme model can be estimated. By concatenating the sub-word phoneme HMMs, a word level HMM can be constructed according to the pronunciation lexicon, thus allowing word level decoding of a speech utterance.

Due to coarticulation effects between two adjacent phonemes, monophone acoustic modelling using only phonemes without any context is too imprecise. The coarticulation effects have to be included in the acoustic model. The amount of context that can be considered depends on the amount of acoustic training data that is available. In the past decade, mainly *triphones* have been applied for acoustic modelling and are still in use e.g. in [Gales2006]. A triphone models a phoneme with left and right coarticulation context, i.e. the phoneme sequence **A B C D** can be modeled using the triphone sequence **$AB ABC BCD CD$** with $ start/end of the sequence. Other state of the art approaches already investigate *quinphones* or even septaphones [Chen2006b]. Using such wide contexts, the amount of parameters that have to be robustly estimated becomes too large: in a septaphone system base on 50 monophones, there exist $(50)^7$ different septaphones. Thus, efficient data-driven or linguistically motivated clustering strategies as described in [Huang2001] are required. Similar phonemes are combined into one linguistic class, thus limiting the amount of free parameters in the system.

3.3.5 Language modelling

To support the decoding process, linguistic information about typical word sequences is required. The following example illustrates the limitations of acoustic modelling:

Mr. **Wright** should **write** to Ms. **Wright right** away.

All marked words share the same pronunciation and can only be distinguished using additional linguistic information. Fixed syntactical rules are not able to cope with the variability of natural language. Both state of the art systems for whole word [Gales2006, Matsoukas06, Chen2006a, Stolcke2006] and sub-word indexing [Larson2003] apply m-gram statistical language models with m>=3 trained on textual data. The vocabulary of the recognizer can be derived from the language model by selecting the most frequent words.

In large vocabulary tasks with a lexicon of 100.000 or more entries, only a small fraction of the possible m-grams (e.g. $(100.000)^3$ different trigrams) can be observed in the training data. Advanced smoothing techniques as reviewed in [Chen1996] must be applied to redistribute the probability mass to unseen m-grams.

The drawback of statistical language models is the huge amount of domain specific data that is required for training (the LVCSR broadcast news transcription system described in [Gales2006] used 1.4 billion words to train a 59k word language model).

3.3.6 Sub-word indexing

Two major disadvantages can be observed when looking at full text transcriptions:

- Language search space complexity. Due to the large amount of words in the vocabulary that is required to gain low error rates, the number of different tri- or 4-grams that have to be estimated is extremely high. Language model pruning as mentioned in [Gales2006] must be applied if the system should work in real-time.

- Closed vocabulary. Out-of-vocabulary words that are not known to the recognizer can never be recognized.

Rather than providing full text transcriptions, some approaches estimate the transcription on a sub-word level. The level of breaking down the words ranges from phoneme based indexing [Ng98, Wechsler1998] to syllable transcriptions [Larson2003]. Sub-word indexing tries to overcome the above mentioned drawbacks of whole word indexing:

- The number of existing sub-word units is much smaller. In German, 5000 syllables are enough for effective syllable-based decoding [Larson2003].

- Although the vocabulary of the indexing unit (e.g. the syllable set) is fixed, the set of words that can be requested during retrieval must not be specified before the recognition process.

On the other hand, more complex retrieval algorithms are required:

- More effort is required to locate keywords in the sub-word transcripts, as the search terms have to be broken down into sub-words first. The sub-word transcription of the search terms can be generated using automatic tools like BOSS [Klabbers2001].

- Due to the smaller acoustic context that is modelled by a sub-word unit, the phoneme or syllable error rate is higher compared to the results from word level approach. In [Larson2003], a search method for fuzzy string matching for compensating the additional recognition errors is proposed, outperforming exact syllable sequence search.

3.3.7 Influence of acoustic environments

It is important to determine the acoustic properties of the audio data that has to be analysed. Although state of the art systems for broadcast news indexing obtain very low speech recognition error rates on planned speech read in a silent environment by a single speaker, the performance degrades under the heterogeneous conditions found in real world data:

- Spontaneous speech, e.g. during an interview.

- More than one speaker at a time.

- Background noise, e.g. during an outside recording at an emergency site.

- Background music.

In the case of VITALAS, no prior assumption can be made regarding the type of data that has to be analysed.

3.3.8 Speaker dependency and adaptability

Unlike speaker dependent recognition systems as found in commercial dictation software, a broadcast indexing system does not have any prior knowledge about the speaker whose speech should be transcribed. Although speaker independent models are trained on a huge amount of acoustic data from a wide range of different speakers, such systems produce significantly higher word error rates on comparable tasks due to different pronunciations, ages or accents.

State of the art systems such as [Gales2006, Chen2006a, Stolcke2006, Matsoukas2006] apply several speaker adaptation techniques for adapting a general speaker independent model to an unknown speaker and thus improve the overall recognition performance.

3.3.9 Language dependency

Unlike the approaches introduced in the section on structural features, speech indexing is a highly language dependent task. In

Table 1, we list the language dependent components of the system that must be adapted if a new language should be explored, along with an estimate for the required effort.

Component	Required Changes	Required Effort
Phoneme Set	New phoneme set must be defined.	High quality phoneme sets already exist for many languages.
Automatic Phonetizer	Language dependent software using linguistic rules, cannot be simply ported to other languages.	Must be bought, freeware tools available for some languages. Must support both phoneme- and syllable transcription of words. In- and output must be converted to match the requirements of the speech recognizer.
Pronunciation Lexicon	Automatic generation of lexicon must be manually corrected. Alternative: lexicon can be bought and converted to required format.	Huge effort for manual correction. High cost for commercial lexica.
Decision Tree for Phoneme Clustering	Must be generated by linguist with knowledge of the new language. Alternative: simple data driven clustering.	High effort for generating the decision tree. Data driven clustering expected to yield lower recognition rates.
Acoustic Models	New Acoustic Models must be trained.	Very high effort: new data must be either bought or collected and the transcriptions must be aligned and converted into the correct

		format.
Language Model	A new language model must be trained.	Very high effort: new data must be either bought or collected. The textual data must be normalized before the statistical analysis can be carried out.

Table 1: Language Dependent Components

The most complex components are the acoustic and language models: knowledge about the language and speech analysis expertise is required to derive the correct transcriptions and train the models. Due to the effort that is needed to collect and prepare the data, the available required linguistic resources are highly expensive (see for example http://www.elra.info/).

3.4 Conclusions

A wide range of different applications in the field of speech analysis has been introduced along with state of the art concepts for coping with the arising challenges in the field of broadcast and TV data, which is an important data source within the VITALAS project.

Mature approaches for structural analysis of broadcast data are available and can be integrated into VITALAS for enriching the annotation provided by the audio module.

For speech indexing, open-vocabulary approaches based on sub-word units yield promising results, overcoming the constraints of whole word transcriptions. Nonetheless, LVCSR systems might prove useful for extracting information that can be used for text mining in a later analysis step.

4 Text indexing

No in depth research regarding text indexing will be achieved in VITALAS project but it is important to select efficient tools adapted to VITALAS textual data. The main objective of this state of the art review is to drive the specification of the text indexing modules of VITALAS system that will be developed in WP2. We will first focus on standard techniques used in most modern text search engines. We will then review more advanced technologies mostly based on word sense disambiguation techniques.

4.1 Standard techniques used in modern text search engines

The objective of this section is to describe standard techniques used in modern search engine that will be implemented in the text indexing module of the WP2.

4.1.1 Information retrieval

Historically, the main issue in information retrieval system was to process text documents since it was the first type of data stored in large databases. Thus, studies have been previously focused on finding efficient ways to select texts from repositories in answer to user requests. Most of the work has been done on the representation of text content and user's queries were naturally simple keywords lists to match those representations. This view of information retrieval system is illustrated in Figure 6 which is a modified view of the scheme proposed by Belkin [Belkin1993].

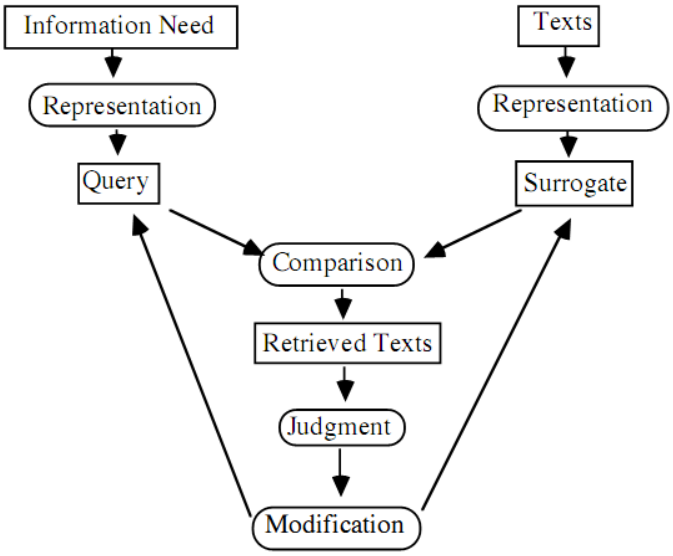


Figure 6 : A model of classic information retrieval approach

Information retrieval systems (IRS) build upon such paradigm have been implemented as a simple function which aims to match user information needs to a set of documents out of a corpus. It is composed of three main steps:

- Extracting a representation of informational content from document and to organize it in a clever way in order to optimize the retrieval process. This is generally called the indexing step of a system when it is applied on a large corpus of documents. Since it involves a large computational cost, it is generally assumed to be the most critical phase in an information retrieval process. Major issues that have to be resolved by this component are formatting, access to the document repositories (or generally to the original content) and information extraction.

- Expressing user's needs in a more-or-less well-specified query. In order to facilitate the similarity evaluation with the document information content representation, it is mainly limited to simple lists of keywords. Additional functionalities have emerged through the different systems implementation and can be used in incremental complexity of query syntax. The most common is the Boolean query which use the AND, OR and NOT operators as a formula to match document content. One can also cite the exact match query linked with term proximity query, the weighted term query, the faceted query (i.e. Limiting the scope of keyword to a certain aspect of document such as “title:IR systems”) or the fuzzy query (searching for words “near” a specified one). But nowadays, **no user's query syntax standard has been proposed for full text query** and only consensus have emerged through the popularity of major web search engines.

- Presenting the list of documents matched with user's needs and providing functionalities to access the document content and if needed, sending updated queries. This is strongly related to user interfaces and thus has a great impact on the user perception of the system capabilities. The general consensus in the proposed information search paradigm is to provide a simple overview on the retrieved documents. The list is ordered by a specific ranking algorithm in order to match the user needs and allow him to quickly choose the documents classified relevant by the system.

4.1.2 Information representation models

Dominik Kuropka [Kuropka2004] proposes a categorization of each model according to two dimensions: the mathematical set of theory involved and the properties of the models. Illustrated in Figure 7, it allows understanding the general incremental process.
In this chapter we will not present all the existing approach but focus on the most important ones: the Boolean model, the Vector Space model and the Probabilistic Model. The implementation of the search engine that will be done on the WP2 of VITALAS will be based on the most robust one.

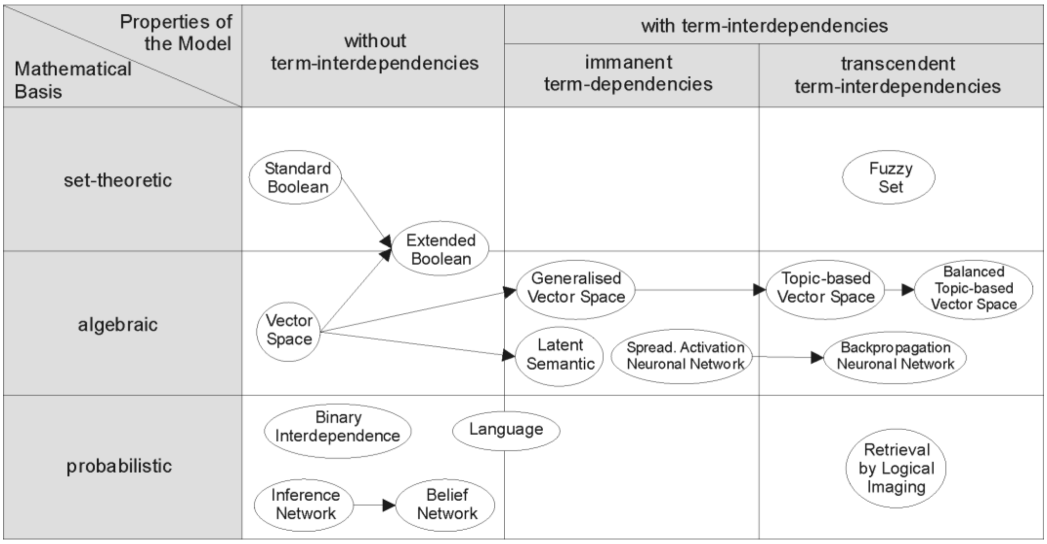


Figure 7 : Evolution of Models for Information Retrieval

The different approaches (regarding mathematical basis) presented in the figure above are:

- **Set-based models** and especially Boolean algebra consider that documents are sets of terms (also called bag of words) and queries are Boolean expressions on terms

- **Algebraic models** use a vectorial representation of information content which can be seen as an extension of the bag-of-words model with a weight assigned on each word. However, this representation in vector or tuples allows using numerous operators and transformations in order to better reflect content of any source of information. Similarity operators can benefit from this enhanced data and thus more advanced evaluation strategies are allowed. The most well known vector model developed by [Salton1975] in the 1970's could be seen matched with the extended Boolean model since the Boolean model had inspired it.
- **Probabilistic models** are involved in more recent studies in order to express the information retrieval process as a multi-stage random experiment where similarities are expressed as probabilities.

The properties of the models presented here are related to terms interdependencies. Natural language processing research field can define this fact through the numerous semantic and grammatical relations which can link terms with each other. The most understandable one is the synonymic relation which express that terms define the same concept. Three levels are defined by Dominik Kuropka:

- Models without terms interdependencies ignore such characteristic of human languages and thus assume that the sense of its terms is independent of the others. As an example in the probabilistic model, it assumes the independence of each probability and in vector space model; it means that each component of the vector is supposed to be orthogonal to the other.
- Models with immanent terms interdependencies allow reducing the dimensionality of a corpus by grouping terms. Thus the model itself fixes the interdependencies directly in accordance to pre-specified assumptions (similarity, edition distance or co-occurrence) generally related to statistical measures.
- Models that allow using external knowledge in order to resolve the dependencies between words. This is the transcendent term interdependency which can involve the most advanced language algorithms

Boolean-based models

As exposed in the previous section most of the current IRS take their roots in the text retrieval field which was the first to tackle information retrieval issues. Thus, the first information representation model, called the Boolean model, was adapted to such textual context. Based on a simple assumption, this model is built on classical Boolean logic. The idea is to look whether or not the keywords from the query are present in the document corpus and then retrieve all documents which contain them.

Numerous optimizations of this model have been developed through the years. Pre-processing are often applied to remove meaningless words. The most common rules is that the lost of words do not have a semantic sense (such as "the" or "from") and thus one is not willing to include them in the retrieval process. The same process has then been applied to the query. In order to reflect this change, the information representation is now assumed to be built on terms with a specific meaning and not simply words.

Almost all current commercial IR systems are based on extensions of the Boolean model because of its simplicity and robustness. This simple model has nevertheless a major drawback since the set of retrieved documents is provided without any ranking mechanism.

Example:

Considering the following document D whose text is "a document that deals with information retrieval but not with information extraction".

Its bag of word representation is d = {a, but, deals, document, extraction, information, not, retrieval, that, with}

Q : ((text U information) ∩ retrieval ∩ ⌐ theory)

In this case the query Q will return the D document.

Vector-space model [Salton1975]:

The vector space model enables to represent each document or query by a vector of weighted terms that are available in the document. It allows using any term frequency measure that has been suggested in information theory (from simple term frequency to advanced term frequency taking into account context information: TFIDF, Pivoted TFIDF [Singhal1996], okapi BM25 [Robertson1998], etc.).

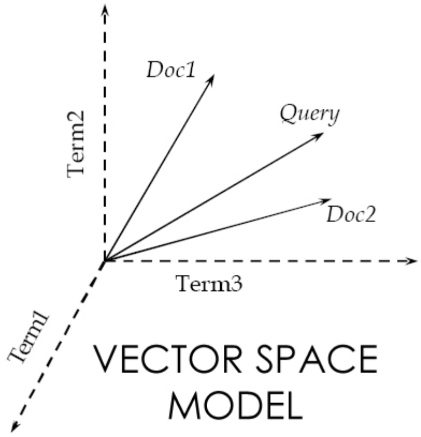


Figure 8 : Salton Vector Space Model

Vector representation of a document (a0,j could be whatever measure of term 0 in document j) :

$$d_j = \{a_{0,j}, \ldots, a_{m-1,j}\}$$

Vector representation of a query (b0 could be whatever measure of term 0 in the considered query):

$$q = (b_0, \ldots, b_{m-1})$$

Several similarity measures exist (Cosine, Jacquard, Dice, etc.) but are always based on a metric between the two vectors. As an example, normalised cosine if defined as:

$$\rho_{cos} = \frac{<\vec{D}_j.\vec{Q}>}{||\vec{D}_j||||\vec{Q}||} = \frac{\sum_{i=0}^{m-1} a_{i,j} b_i}{\sqrt{\sum_{i=0}^{m-1} a_{i,j}^2} * \sqrt{\sum_{i=0}^{m-1} b_i^2}}$$

Example:

The vector space representation of the document is :

Term-Space = d {a, but, deals, document, extraction, information, not, retrieval, text, that, theory, with}

d = (1,1,1,1,1,2,1,1,0,1,0,2)

Q= text information retrieval theory
q=(0,0,0,0,0,1,0,1,1,0,1,0)

ρ_{cos}*=3/(4*2)=0.375*

In this case, the query Q will return the document D with a score of 0.375

<u>Probabilistic Model</u> [Sparck1998]:

This model is based on the idea that a retrieval function can be seen as the probability that a document is relevant for a given query. The rank function is then defined as a function based on the combination of the probability of a document relevance and non relevance conditionally to a document and a query.

Typically the retrieval status function (rank function) is:

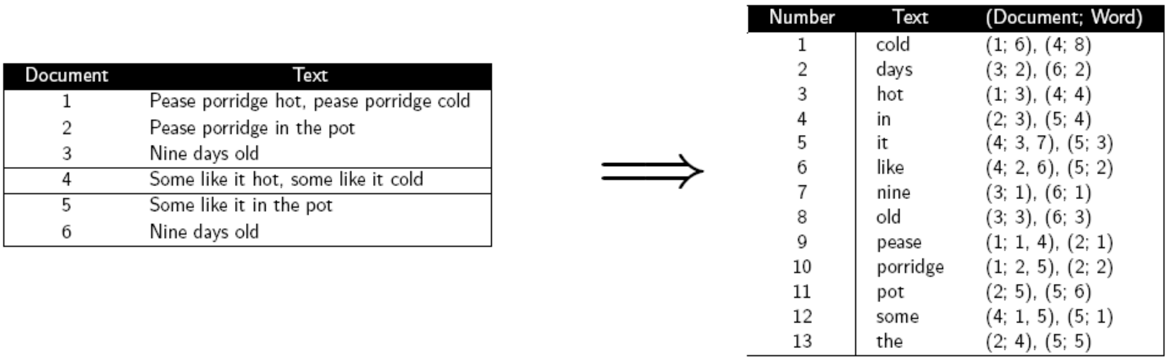
$$RSV(D_j|Q) \quad = \quad \log \frac{P(R|D_j, Q)}{P(\neg R|D_j, Q)}$$

The documents are then ranked according to the value of the retrieval status function.

4.1.3 Indexing structure

Inverted lists are today the most common indexing structures for texts as they enable fast and efficient searching mechanisms. There are based on the conversion of the documents into a set of records associating each term of the documents set to the documents in which it is present. Faceted indexing is done by storing inverted list also for metadata, where metadata are keys.

During the evaluation, searching consists in traversing lists for each query term. The evaluation of Boolean operator consists then in Boolean logics on results (OR: the union of matching documents, AND: an intersection of matching documents, Proximity, etc.). Other data are often stored in the inverted list in order to perform more advanced queries. Most popular additional data is for instance the positions of the terms in the documents.



Document	Text
1	Pease porridge hot, pease porridge cold
2	Pease porridge in the pot
3	Nine days old
4	Some like it hot, some like it cold
5	Some like it in the pot
6	Nine days old

$\Longrightarrow$

Number	Text	(Document; Word)
1	cold	(1; 6), (4; 8)
2	days	(3; 2), (6; 2)
3	hot	(1; 3), (4; 4)
4	in	(2; 3), (5; 4)
5	it	(4; 3, 7), (5; 3)
6	like	(4; 2, 6), (5; 2)
7	nine	(3; 1), (6; 1)
8	old	(3; 3), (6; 3)
9	pease	(1; 1, 4), (2; 1)
10	porridge	(1; 2, 5), (2; 2)
11	pot	(2; 5), (5; 6)
12	some	(4; 1, 5), (5; 1)
13	the	(2; 4), (5; 5)

Figure 9 : Inverted list structure

Many access methods to crawl the index efficiently can be found in the literature. Among others, we can quote B*-tree [Berliner1979] and Hashing functions. One-dimensional index structures are briefly introduced in section 5.2.3.1of this document.

4.1.4 Scalability issue: distributed indexing

As the volume of documents grows, it is sometimes not possible to carry out the indexing on a single machine. It's typically the case for web indexing and in this case the indexing is distributed over a cluster of machines such as the MapReduce architecture depicted below:

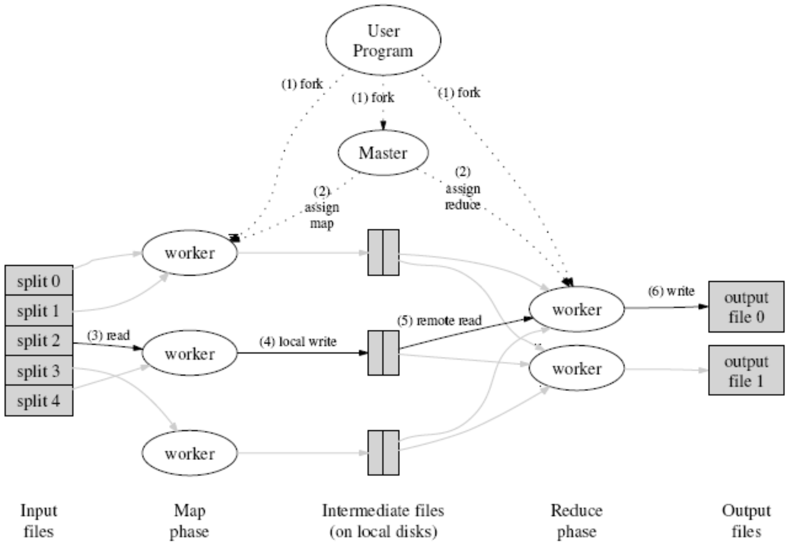
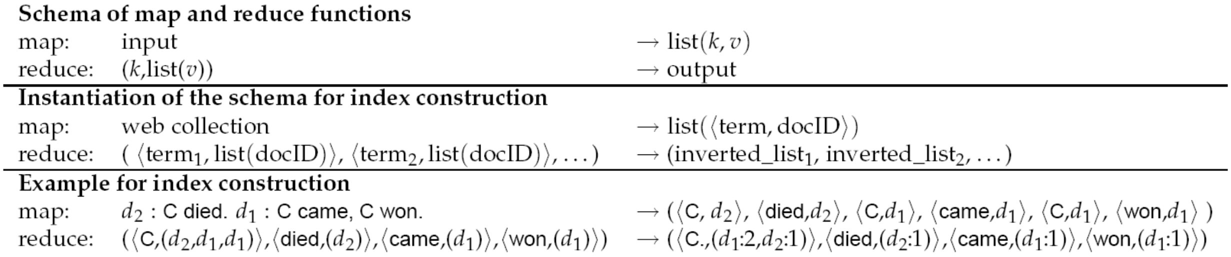


Figure 10 : The map reduce architecture

The main principle of distributed indexing as described in [Dan2004] is to have a master node that divides the work up into tasks that can be distributed to worker nodes.
The MapReduce procedure simplifies a large problem into smaller problems. It is based on two distinct phases called Map and Reduce:

- Map: the first part of the algorithm consists in splitting the tasks into key-value part (the name map is due to this function). At indexing time, the map phase consists in transforming the document parts into a map of key-value pairs (usually term-document).
- The reduce phase, will then consist in merging all these key-value pairs into a global structure.



Schema of map and reduce functions		
map:	input	$\rightarrow$ list(k, v)
reduce:	(k,list(v))	$\rightarrow$ output
Instantiation of the schema for index construction		
map:	web collection	$\rightarrow$ list($\langle$term, docID$\rangle$)
reduce:	($\langle$term$_1$, list(docID)$\rangle$, $\langle$term$_2$, list(docID)$\rangle$, ...)	$\rightarrow$ (inverted_list$_1$, inverted_list$_2$, ...)
Example for index construction		
map:	d_2 : C died. d_1 : C came, C won.	$\rightarrow$ ($\langle$C, $d_2\rangle$, $\langle$died,$d_2\rangle$, $\langle$C,$d_1\rangle$, $\langle$came,$d_1\rangle$, $\langle$C,$d_1\rangle$, $\langle$won,$d_1\rangle$)
reduce:	($\langle$C,(d_2,d_1,d_1)$\rangle$,$\langle$died,(d_2)$\rangle$,$\langle$came,(d_1)$\rangle$,$\langle$won,(d_1)$\rangle$)	$\rightarrow$ ($\langle$C.,(d_1:2,d_2:1)$\rangle$,$\langle$died,(d_2:1)$\rangle$,$\langle$came,(d_1:1)$\rangle$,$\langle$won,(d_1:1)$\rangle$)

Figure 11 : Schema of Map and Reduce functions (from [Bayer1972])

The index is then distributed across a large cluster of machines for supporting querying. Two obvious alternative index implementations are suggested by the index structure itself:

- partitioning by terms, also known as global index organization

- partitioning by documents, also know as local index organization

In the former, the dictionary of index terms is partitioned into subsets, each subset residing at a set of nodes. Along with the terms at a node, the list of documents and term's measurements for those documents are kept. A query is routed to the nodes corresponding to its query terms and results came easily.

In principle, this allows greater concurrency since a stream of queries with different query terms would hit different sets of machines.

However in practice, partitioning indexes by dictionary terms turns out to be unwieldy even though the partionning of query terms is optimized by the use of co-occurrences of query terms and entails the clustering of terms. The problem is that multi-word query will need to exchange long posting list out of the index nodes to later merge them and reduc the number of documents in the result set. Moreover in dynamic index contex adding a new document will have an impact on many node and thus induce much more difficulty. Thus the most efficient implementation is the document partitionning: each node contains the index for a subset of all documents. When queried, the system will distribute the query on all node but the results set answered will be far smaller than in the query partitioning implementation and thus the global network overload will be will not be overloaded. This strategy trades more local disk seeks for less inter-node communication. One difficulty in this approach is that global statistics used in scoring – such as idf – must be computed across the entire corpus even though the index at any single node only contains a subset of the documents.These are computed by distributed "background" processes that periodically refresh the node indexes with fresh global statistics.

The strategy use to partition the documents corpus must then be wisedly chosen. A common technique in web-based index is to distribute the documents out of the same domain name on the same node. A similar strategy could be involved in VITALAS assuming that each source of content (i.e. content providers storage system) can be considered as a domain. It could also be based on the crawler distribution. But deeper experiments have to be achieved in order to define an optimized strategy to limit query response time.

4.2 Word sense disambiguation techniques

4.2.1 Word sense disambiguation (WSD)

Many words have several meanings or **senses**. A word is semantically ambiguous if it has more than one sense. In a clear-cut scenario [Manning1999] we might have two clearly distinct senses of a word such as **bank**:

- the rising ground bordering a lake, river or sea …
- an establishment for the custody, loan exchange, or issue of money …

The task of disambiguation is to determine which of the senses of an ambiguous word is invoked in a particular use of the word. A word is assumed to have a finite number of discrete senses, often given by a dictionary or thesaurus, and the task of the Word Sense Disambiguation (WSD) program is to choose between these. However, it is not always clear where to draw the line between senses, as seen in the following overlapping senses of the word **title** [Manning1999]:

- Name/heading of a book, statue, work of art or music, etc.
- Material at the start of a film
- The right of legal ownership of land
- The document that is evidence of this right
- An appellation of respect attached to a person's name
- (the whole) written work

Dictionaries differ in the number and kind of senses they list. A word can also be used as different parts of speech, e.g. **butter:**

- bread and butter (noun)
- you should butter (verb) your toast.

Determining the usage of a word in terms of part of speech (POS) is known as **tagging** [Manning1999]**.**

WSD is useful in a number of areas of text processing:

- Information retrieval, where we only want documents on *bank* as a financial institution
- Machine translation, such as the translation of queries for cross-language information retrieval (CLIR) - do we translate *bank* as *Ufer* or *Bank?*
- WSD is necessary for automatic thesaurus generation, as different senses of a word should be placed in different parts of the thesaurus.

WSD is done by looking at the **context** of the word's use. Since the surface form of the word in all its sentences is identical, we cannot disambiguate its sense by looking at that word alone, but we must look at the words and phrases which surround the word we are trying to disambiguate. In general nearby clues are useful for determining POS, (for example, is the previous word a determiner?) while a broader context is more useful for semantic disambiguation [Manning1999].

Having defined WSD, in section 4.2.2 we will examine the range of individual techniques which have successfully been used for the purpose. These techniques are a) to assume that the most frequent sense of a word is always correct, b) WSD is facilitated if we know what part-of-speech our ambiguous word takes, c) only one sense of a word is found in each discourse or collocation, d) a number of machine-readable resources have proved valuable in WSD, namely dictionaries, thesauri, bilingual parallel corpora and idiom lists. We will then move on to machine learning techniques, both supervised learning, where the system learns from examples of humanly sense-tagged text, and unsupervised learning where the system learns from the various patterns of co-occurrence of ambiguous words with other linguistic features that some words take more than one sense. One family of unsupervised learning techniques is graph-based methods. A number of successful systems such as TiMBL and ACAMRIT have used a combination of these approaches in stepwise fashion. In section 4.2.3 we will discuss some of these systems. In section 4.2.4 we will consider how WSD approaches have been compared with each other, and finally, in section 4.2.5, we will discuss issues of scalability, since the WSD modules to be built by the VITALAS project must achieve broad coverage (work for a very wide range of words) and overcome problems of obtaining sufficiently large amounts of training data (the knowledge acquisition bottleneck). We will conclude with a summary of recommendations for the VITALAS project based on the findings of this report.

4.2.2 Techniques for WSD

- The simplest approach, making use of word sense dictionaries with frequency information, is to assume that a word is always used in its **most common sense**. Thus if the word *list* commonly refers to a collated sequence as in a shopping list, and only very rarely refers to blowing as in "the wind listeth", we assume that the first sense is always correct.
- Part of speech tagging. In section 1 we looked at the example of *butter*, which has at least two related senses according to whether it is a noun or a verb. Other words, such as *lead*, have completely different senses when they take different parts of speech; *lead* as a noun is a metal, while *lead* as a verb means to go first so others can follow.

- Disambiguation based on machine-readable resources, such as dictionaries, thesauri, bilingual parallel corpora and idiom lists.
- **Supervised** disambiguation, based on a labelled training set, with examples of words previously assigned sense labels, from which the sense labels of words in new texts may be inferred.
- **Unsupervised** disambiguation, where only unlabelled text corpora are available for training. Techniques include factor analysis and clustering

4.2.2.1 Baseline: most frequent sense

The simplest approach, often taken as the baseline in evaluation experiments, is the general likelihood of a word taking a particular meaning, as found in certain frequency dictionaries. For example if a corpus has 5651 occurrences of *bridge* in the sense of a bridge over a river, and only 194 occurrences of *bridge* in the sense of a dental bridge, then the simplest technique is to assume that the more common sense of "bridge over a river" is always the correct one. According to Allen [Allen1995], this simple technique is about 70% accurate over English as a whole.

4.2.2.2 Part-of-speech tagging

Wilks and Stevenson [Wilks1996] have shown that POS tagging (automatically labelling each word in a text according to the part of speech it takes) greatly assists in the problem of word sense disambiguation. For example, if *spring* is a verb, we know it must mean *jump*. They found that 95% of word types in the Longman Dictionary of Contemporary English (LDOCE) could potentially be disambiguated if we know the POS they take in a given context. They also found over 42% reduction in error rate when using part-of-speech based disambiguation alone compared with a baseline of simply choosing the most frequent homograph (LDOCE word senses are sorted in descending order of frequency). Using part-of-speech filtering of word senses is a safe method which is unlikely to reject the correct sense [Stevenson2001]. Automatic part of speech taggers include the probabilistic CLAWS tagger [Garside1987] and the rule-based Brill tagger [Brill1995].

More recently, Toutanova et al. [Toutanova2003] developed a new part-of-speech tagger that demonstrates the following ideas: (i) explicit use of both preceding and following tag contexts via a dependency network representation, (ii) broad use of lexical features, including jointly conditioning on multiple consecutive words, (iii) effective use of priors in conditional log-linear models, and (iv) fine-grained modelling of unknown word features. Using these ideas together, the resulting tagger provides a significant error reduction on the best previous single automatically learned tagging result.

4.2.2.3 One sense per discourse, one sense per collocation

The dictionary-based algorithms we have looked at so far process each occurrence of a word separately. However, we can also consider constraints between different occurrences that can be exploited for disambiguation [Manning1999]. One such constraint is Yarowsky's notion of **one sense per discourse** [Yarowsky1995]. The sense of a word is highly consistent within any given document. For example, if we find one occurrence of **plant** used in the sense of a **living organism,** the word is likely to keep this meaning right through the article, where it is unlikely ever to be used for **industrial plant.** Yarowsky's second constraint is **one sense per collocation.**

Collocations are pairs or groups of words that frequently appear in the same context. Are any collocates of the word, suggesting a particular interpretation found in its immediate vicinity? We would, for example, prefer the dental sense of *bridge* if the context contains collocates such as *dentist* or *cavity*. This technique is also called proximity disambiguation. The amount of text on either side of a word in which we look for collocates is called the window. One statistical measure of collocation strength is Mutual Information [Church1990]. This is a function of how often do two events occur together (e.g. we find *kith* and *kin* in the same window), compared with how often these two words occur in the data set as a whole, expressed by the following formula:

$$MI(x,y) = \log_2 \left[\frac{p(x,y)}{p(x).p(y)} \right]$$

Collocation filters have been used by various authors, the main variations being in a) the window length, b) whether to measure word-word, word-phrase, or noun-argument collocations. In order to calculate collocation statistics for verbs and the nouns they are associated with, shallow parsing is required beforehand. Argaw [Argaw2005] makes use of collocation filters to help translate search engine queries from Amharic into French.

Rule-based collocation filters can be in the form of **decision trees.** According to the nature (word classes or individual words) of the neighbours of the test word, decisions are taken, in an order designed to yield maximum information at each branch point, and to determine the sense of the test word in as few steps as possible. A similar technique for French grammatical words is described by Hug [Hug2000].

4.2.2.4 Use of machine-readable resources

4.2.2.4.1 Machine-readable dictionaries

WSD can be enabled by matching dictionary example sentences with a window of input text. This is based on the simple idea that a word's dictionary definitions are likely to be good indicators for the senses they define [Manning1999], as in the example of **cone,** which may be defined in at least two ways:

- a mass of ovule-bearing or pollen-bearing scales or bracts in **trees** of the pine family
- a crisp wafer for holding **ice** cream

These definitions suggest that whenever the context of *cone* contains *tree*, it means pine-cone, and whenever it contains *ice,* it means ice-cream-cone. In the simplest case, a score is given for the number of identical words (after stop-listing and stemming) occurring in the dictionary example and the text window in which the word we are trying to disambiguate occurs. The sense with the highest scoring example is chosen. Manning and Schütze [Manning1999] give the example of **ash:**

- **Sense 1 (tree):** a tree of the olive family
- **Sense 2 (burned stuff):** the solid residue left when combustible material is burned.

Score (Sense 1)	**Score (Sense 2)**	**Context**
0	1	This cigar burns slowly and creates a stiff ash
1	0	The ash is one of the last trees to come into leaf

Various measures of similarity between test sequences and dictionary examples (glosses) have been suggested. The earliest example of this approach was described by Lesk [Lesk1986], who was interested in the degree of overlap in two dictionary definitions, as in the following example, where sense 1 of *pine* corresponds with sense 2 of *cone* in the phrase *pine cone*:

- Pine
    - 1. kinds of **evergreen tree** with needle-shaped leaves
    - 2. waste away through sorrow or illness
- Cone
    - 1. solid body which narrows to a point
    - 2 fruit of certain **evergreen trees**

Cowie et al. [Cowie1992] enhanced this approach with a machine learning technique called simulated annealing. Like Lesk, they employ the rationale that "word senses which belong together in a sentence will have more words in common in their definitions that word sentences which do not belong together". Unlike Lesk, they do not test all possible configurations of every sense of every word in the sentence, but use simulated annealing as a heuristic search to find the combination of word senses which has the least "energy", where energy E is a measure of the degree of overlap in the dictionary definitions. First redundancy R is computed by giving a stemmed word form which appears n times (across all definitions) a score of n-1 and adding up the scores. E is then 1 / (1 + R). The starting configuration is the most likely word sense for each word, as given in LDOCE). The initial value of E is calculated, and then we randomly replace one word sense with another for the same word. If E falls, keep the change, but if E rises, make the change with probability P as given by the following formula:

$$P = e^{(\Delta E / T)}$$

This formula means that moving to a "poorer" configuration (with lower E) becomes less likely as time goes on, but is more likely if the increase in energy is slight. Eventually the technique converges on the (hopefully) optimal configuration. Simulated annealing is less likely to get stuck in a local minimum than a greedy search (which would only ever move to a configuration with lower energy).

Cowie et al [Cowie1992] contrast such numerical techniques with semantic techniques, which are based on linguistic information individual to each ambiguous word. These require extensive hand-crafting of rules, involving such tasks as assigning semantic categories to nouns, or semantic preferences [Wilks1973] to verbs and adjectives.

4.2.2.4.2 Thesauri

Thesauri, hierarchies of words and the relations between them have been used in WSD to overcome data sparseness. A recurring problem with WSD, compared with POS tagging, is that there are more word senses than syntactic categories, meaning a much larger amount of training data is required. Use of a thesaurus helps overcome this problem, as frequencies of word classes are studied rather than those of individual words. One can thus count matches which involve semantically related words (such as words with the same Roget's thesaurus categories) in the matching score, rather than insisting on exact word matches.

The use of a thesaurus also enables the use of expert system-like inference, such as spreading activation [Sussna1993] [Voorhees1993]. It also enables the notion of conceptual density. All possible senses of all content words in the input sentence are marked in a hierarchy such as WordNet [Fellbaum1998]. The portion of the hierarchy with the greatest concentration of marked nodes (including one for the word being tested) will reflect the sense of the test word.

Semantics-free, word-based information retrieval has two problems: low precision when we retrieve docs pertaining to other senses of the word (polysemy/false positives/low precision), low recall when relevant items are not indexed by the search term, but by alternative keywords (synonymy / false negatives/ low recall). Sussna [Sussna1993] makes use of semantic distance between network nodes - the network being WordNet (a network of word meanings connected by a variety of lexical and semantic relations, which contains over 35000 word meanings represented as network nodes called "synsets" or synonym sets). Each sense of a word connects to a different synset: the relations used are synonymy (all members of a node have the same word sense); hypernymy (is a); hyponymy (has instance); holonymy (is part of); meronymy (has part/contains); and antonymy (is complement of - a self inverse relation). "Our approaches to WSD are based on minimising an objective function using the notion of semantic distance between synsets in WordNet".

Edge weightings take into account the number and type of steps and the density or fan-out of the tree at that point. (Sussna used weights of 0 for synonymy, 2.5 for antonymy, 1 to 2 for other word relations). The technique is to find the shortest path connecting the nodes. "The hypothesis is that given a set of terms occurring near each other in the text, each of which might have multiple meanings, by picking the senses that minimise distance we select the correct senses" [Sussna1993]. For example, the *tree* sense of *pine* would be close to the *tree* sense of *cone* (perhaps separated only by the part_of relation), while the *pine_away* sense of *pine* would be quite distant. Vorhees [Voorhees1993] also used WordNet for WSD, making use of its relations.

Sussna [Sussna1993] also wrote that the use of a thesaurus enables the notion of conceptual density. All possible senses of all content words in the input sentence are marked in a hierarchy such as WordNet [Fellbaum1998]. The portion of the hierarchy with the greatest concentration of marked nodes (including one for the word being tested) will reflect the sense of the test word. A WordNet demonstration is available at http://vancouver-webpages.com/wordnet .

4.2.2.4.3 Bilingual parallel corpora

Gale, Church and Yarowsky [Gale1992] used machine readable texts and their translations for WSD, noting for example that the sense of drugs which translates into French as medicaments collocates with prescription, patent and generic while the sense which translates as drogues collocates with abuse, paraphernalia and illicit. Other authors who have used bilingual corpora in WSD are Dagan et al. [Dagan1991] and Brown et al. [Brown1991].

Dagan et al.'s paper presents an approach for resolving lexical ambiguities in one language using statistical data on lexical relations in another language. They not the similarities between WSD and the selection of the most appropriate target word in the translation of a source language word. For example, the following transliterated Hebrew and English sentences are correct translations of each other:

- Nose ze mana' mi-shtei ha-mdinot mi-lahtom 'al hoze shalom
- That issue prevented the two countries from signing a peace treaty.

Lahtom has four possible senses: *sign, seal, finish, close*, while *hoze* has two: *contract* and *treaty*. Using corpus data (single word frequencies for English), and a naïve approach of choosing the most frequent word yields the following interpretation of the original Hebrew sentence:

- That issue prevented the two countries from closing a peace contract.

A better solution was to identify the lexical relationships in the corpora of the target language, instead of the source language. This involved parsing the corpus, and then counting how often words appear in the same syntactic relation in the whole corpus as in the ambiguous sentence. The relation *peace-treaty* appeared 49 times, and *peace-contract* never, suggesting *treaty* is more likely to be the sense of *hoze* than *contract*. Relations involving *treaty* occurred the following numbers of times: *sign-treaty* 79, *seal-treaty* 0, *finish-treaty* 0, and *close-treaty* 0, showing that *sign* is the likeliest sense of *lahtom.*

Brown et al [Brown1991] describe another method of statistical word alignment in a parallel corpus. A pair of aligned words is called a connection. Using *p(e,f)*, the proportion of aligned word pairs which contain both the French word and the English word, *p(e)*, the proportion of aligned word pairs containing the English word, and *p(f)*, the proportion of aligned word pairs containing the French word, we can compute the mutual information between a French word and its English "mate" in a connection. Their paper describes a method for labelling a word with a sense that depends on the context in which it appears in such a way as to increase the mutual information between the members of a connection. Compare the following translations:

Je vais **prendre** ma decision → I will **make** my decision.

Je vais **prendre** ma voiture → I will **take** my car.

We can assign a sense to *prendre* (make or take) by seeing whether the first noun to the right of *prendre* is *decision* or *voiture*. We call the noun to the right the "informant" for *prendre*.

Given such a potential informant, we let *f* be a French word and let English words which are possible translations of *f* be divided into two classes. The problem is to find a division of the English translations of *f* into two sets which has maximal mutual information with two corresponding sets of French informants. With the flip-flop algorithm one alternates between splitting the French informants into two sets and the English translations of *f* into two sets. A technique called the splitting theorem finds the division of the two vocabularies (French informants and English translations of *f*) which yields maximum total mutual information. For example, if the sets of French informants are denoted as *a* and *c*, and the sets of English translations of *f* are denoted *b* and *d*, a being matched with *b* and *c* being matched with *d*, we might determine the overall mutual information as follows:

a = [null | mesure | note | exemple | temps |initiative | part]

b = [make]

c = [decision | parole | connaisance | engagement | fin | retraite]

d = [take]

$$MI = \log_2\left[\frac{(a,b)}{(a)(b)}\right] + \log_2\left[\frac{(c,d)}{(c)(d)}\right]$$

This formula describes the situation where we have two senses of *prendre*, but can be extended to accommodate more word senses. Seven types of types of informants were considered: word to the left; word to the right; first verb to the left; first noun to the right; first verb to the right; noun to the left; tense of the current word for nouns.

4.2.2.4.4 Idiom lists

Idiom lists are useful in WSD, since the sense intended by an idiom as a whole is deemed to be more likely than the senses of the individual words contained within it. Thus if someone is "glued to his seat", he is transfixed, but probably not by the application of glue. Idiom extraction patterns can be created manually using scripting languages such as Perl. The example "shake in \w* (shoes|boots|seat)" would cover any of the variable form idioms "shake in your boots", "shake in one's seat", etc.

4.2.2.4.5 Supervised learning for WSD

With supervised WSD, a machine learning technique, a disambiguated corpus is available as the **training** set, where each occurrence of an ambiguous word is annotated with a semantic (sense) label. The task is to build a classifier which classifies new cases (**test** set) based on their context of use.

Since manual sense tagging is laborious, **pseudowords** can be used - all occurrences of two quite distinct words are artificially conflated, to see whether their original meanings can be recovered by the WSD algorithm [Sanderson2000]. For example, all occurrences of **banana** or **door** in a corpus can be replaced by **banana-door**. This overcomes the need for hand labelling: the text with pseudowords is regarded as the ambiguous source text, and the original is regarded as the same text with the ambiguity removed.

The unsupervised approach of Diab and Resnik [Diab2002] utilizes parallel corpora for word sense tagging. They carried out an investigation of the feasibility of automatically sense annotating large amounts of data in parallel corpora using an unsupervised algorithm, making use of two languages simultaneously only one of which has an available sense inventory. The method aims at achieving two main goals: first producing large quantities of reasonably if not perfectly sense annotated data for the language with the sense inventory in order to bootstrap supervised learning techniques without the need for manual annotation, and secondly achieving sense tagging using that same sense inventory for the second language, thus creating a sense tagged corpus and automatically making a connection to the first language's sense inventory.

For example in a French-English corpus, the French word “catastrophe” could be found in correspondence to English “disaster” in one instance and to “tragedy” in another. Each of those English words is itself ambiguous e.g. “tragedy” can refer to a kind of play as opposed to “comedy”, and also to refer to a disastrous situation in real life. We can take advantage of the fact that both English word instances appear in correspondence with “catastrophe” to infer that they share some common element of meaning and we can use that inference in deciding which of the English senses was intended. Having done so we can go further, we can project the English word sense chosen for this instance of “tragedy” to the French word “catastrophe” in this context thus tagging the two languages in tandem with a single sense inventory. The approach of Diab and Resnik [Diab2002] is decomposed into the following steps:

1) Identify words in the target English corpus and their corresponding translations in the source French corpus.
2) Group the words of the target language forming target sets that were translated into the same orthographic form in the source corpus.
3) Within each of these target sets consider all the possible sense tags for each word and select sense tags informed by semantic similarity with the other words in the group.
4) Project the sense tags from the target side to the source side of the parallel corpus.

The evaluation of Diab and Resnik's unsupervised WSD approach [Diab2002] showed that its accuracy was comparable or superior to most other unsupervised systems.

Stokoe et al. [Stokoe2003] describe a sense-based vector space retrieval model for information retrieval. They introduced the SF*IDF measure where a word sense is typical of a given context if it is both frequent in that context and relatively rare in all other concepts. Stokoe's own algorithm worked in a stepwise fashion, where firstly all possible senses of a word are found, then we examine the context of a particular word to disambiguate. Does the context a) contain any immediate left or right collocates identified from the Semcor corpus, or b) contain any collocations (not necessarily immediate) associated with a given word sense as found in Semcor? If these examinations of context fail to yield an answer, then WordNet frequency statistics are used for the default estimation.

Chen [Chen2006b] used novel methods to extract linguistic features in the contexts of ambiguous verbs, in order to build up feature vectors for supervised WSD with various clustering models. These linguistic features were the verb's subject, semantic features associated with pronouns, common nouns and proper nouns, and sentential complements of verbs.

Publicly-available supervised WSD systems include SenseTools, SyntaLex and WSD Shell. The limitations of these systems are the lack of off-the-shelf components for integration with other sources such as WordNet, and they do not make use of such features as named entities (such as lists or gazetteers of persons, places, drugs or components), coreference resolution (to recognise for example that "the city" refers to "Sunderland") and morphological forms (recognising for example that different grammatical variants belong to the same root word). Joshi et al.'s own solution [Joshi2006a] was to combine Sheffield University's GATE (General Architecture for Text Engineering) system for the identification and extraction of linguistic features, the NSP (N-gram Statistics Package) for selection of the most promising features for WSD, and the WEKA (Waikato Environment for Knowledge Analysis) environment which allows the selection of suitable algorithms for the classification of feature vectors into word sense categories. In a separate study [Joshi2006b], they used a Support Vector Machine as their classifier.

McCarthy et al. [McCarthy2004] presented work on the use of a thesaurus acquired from raw textual corpora and the WordNet similarity package to find predominant noun senses automatically. In order to find the predominant sense of a target word they used a thesaurus acquired from automatically parsed text based on the method of Lin [Lin1998]. This provides the nearest neighbours to each target word, along with the distributional similarity score between the target word and its neighbour. They then used the WordNet similarity package [Patwardhan2003] to give a semantic similarity measure (referred to as the WordNet similarity measure) to weight the contribution that each neighbour makes to the various senses of the target word.

Martinez et al. [Martinez2006] proposed a novel disambiguation algorithm that improves previous systems based on acquisition of examples by incorporating local context information; this was applied using "monosemous relatives" approach. The unsupervised systems that were employed in this work require raw corpora and a thesaurus with relations between word senses and words. Although these resources are not available for all languages, there is a growing number of WordNets in different languages that can be used. The "monosemous relatives" approach is a technique to acquire training examples automatically and then feed them to a Machine Learning (ML) method. This algorithm is based on Leacock et al. [Leacock1998], and follows three steps: (i) select a set of monosemous words that are related to the different senses of the target word, (ii) query the Internet to obtain examples for each relative, (iii) create a collection of training examples for each sense, and (iv) use an ML algorithm trained on the acquired collections to tag the test instances.

In this work, the monosemous relatives are obtained using WordNet, and different relevance weights are assigned to these words depending on the distance to the target word (synonyms are the closest, followed by immediate hypernyms and hyponyms). The goal of this new approach was to use the WordNet relatives and the contexts of the target words to overcome some of the limitations found in the "monosemous relatives" technique. One of the main problems is the lack of close monosemous relatives for some senses of the target word. This forces the system to rely on distant relatives whose meaning is far from the intended one. Another problem is that by querying only with the relative word we do not put any restrictions on the sentences we retrieve. The basic idea of the "relatives in context" method is to combine local contexts of the target word with the pool of relatives in order to obtain a better set of examples per sense. Using this approach, we only gather those examples that have a close similarity with the target contexts, defined by a set of pre-defined features.

4.2.2.5 Unsupervised learning

Supervised learning is based on the fixed-list of senses paradigm, where the possible senses of words are a closed list derived from a dictionary or lexicon. Even though lexicographers and semanticists have long warned about the theoretical problems of such an approach, supervised WSD has performed quite well in open evaluation exercises [Snyder2004]. Unfortunately supervised learning requires large amounts of training data to yield reliable results, and its coverage is limited to the words for which sense-labelled data exists. Sometimes no training data is available, such as when we are working in specialised technical domains, or working with search engines which must be effective for all domains. In such cases supervised learning is impossible, and we must consider unsupervised learning. While supervised learning approaches are trained on manually sense-tagged text, with unsupervised learning the classification of the data (set of tags) is not known beforehand [Argaw2005]. Yarowsky [Yarowsky1995] defines sense tagging as "assigning each instance of a word to established sense definitions (such as in a dictionary). This differs from sense induction: using distributional similarity to partition word instances into clusters that may have no relation to standard sense partitions". Strictly speaking, completely unsupervised disambiguation is not possible if we mean **sense tagging** (an algorithm that labels occurrences as belonging to one sense or another). Sense tagging requires that some characterisation of the senses be provided. However **sense induction** can be performed in an entirely unsupervised manner: one can cluster the contexts of an ambiguous word into a number of groups and discriminate between those groups without labelling them.

Agirre, de Lacalle and Martinez [Agirre2006b] show that the more feature types one considers in a disambiguation algorithm, the better are the results. Having said this, some feature selection is still necessary as this will reduce noise resulting from redundant information. They considered two combinations of classifiers: a) a simple vote between k-Nearest Neighbour (kNN), a vector space model and Joachims' SVMLite [Joachims1999], and b) a combination of kNN classifiers trained on different partitions of the feature set. The linguistic feature types were grouped into three main sets: local collocations: a) bigrams and trigrams found within a set number of words around the target (the word to be disambiguated). Such n-grams were constituted by lemmas, word forms or POS tags; b) syntactic dependencies, which were regular expressions defined by the POS tags around the target, i.e. object, subject, noun-modifier, preposition and sibling; c) global features based on the occurrence of lemmas of content words both throughout the corpus and within a plus-or-minus four word window around the target.

For both the kNN classifier and the vector space model, the similarity between a test pattern and training instances (both represented by feature vectors) was measured by the cosine of their vectors. When using the vector space model, centroids were obtained for each sense of a word in training - in the test phase, they looked for the closest centroid. When using kNN, the closest class was given by a weighted vote of k nearest vectors, where the nearest neighbour had a voting weight of 1, the second nearest neighbour had a voting weight of ½, the third nearest was given a voting weight of 1/3, and so on. Best results were obtained with k-NN.

Biber [Biber1993] used the multivariate statistical technique of **factor analysis** to discover four basic senses of the word "right" in a corpus, according to their various collocates. The inputs to his factor analysis program were the frequency counts of each collocation of "right" or word-pair containing "right" in each subtext of a 30-million word corpus. The idea was that collocations referring to the same sense of "right" would tend to group together in similar subtexts. Factor Analysis identifies the group of collocations found together that explains most of the variance in the data, and calls these the first factor. This consisted of collocations such as "right hemisphere" or "right ear", all clearly referring to the right hand side. The second factor, explaining the next most variation in the data, grouped such collocations as "right there" and "right back", where "right" was in the sense of "directly" or "exactly". Two more factors were found, one with such collocations as "all right" and "that's right" with the sense of "OK or correct", and one corresponding to a less clear cut sense, containing among others "right you" and "right so". Using the same technique, he found that "certain" meant "particular" in some contexts (as in "certain other" and "certain extent"), and "sure" in others (as in "certain that", or "make certain").

Wang and Carroll's approach [Wang2005] provides only single word translations, and proceeds as follows:

1. Translate an English ambiguous word (in each of its senses) to Chinese using a lexicon with word sense information. Assume each sense has a distinct Chinese translation.
2. Construct a corpus of text snippets, word segmented, retrieved either using a search engine or a large Chinese corpus in response to queries of each of the Chinese translations of the ambiguous word.
3. Use a Chinese-English dictionary to translate word-for-word the Chinese text snippets, in each case yielding a back-translated English ambiguous word and context information pertinent to one particular sense. This is an example of the bag-of-words model, since we retain no ordering and sequence information.

We can go beyond the bag-of-words model, enabling the retention of other linguistic features useful for classification, such as word order, part-of-speech sequence, by replacing the lexicon with a machine translation (MT system). An MT system also reduces the long-term costs of generating manually sense-tagged data. Although an MT system itself is expensive to build, it can be used repeatedly to generate as much data as is required. The system was successfully demonstrated at Senseval-3 [Mihalcea2004].

Wang and Martinez [Wang2006b] also used an unsupervised approach to WSD, thus avoiding the reliance on manually annotated resources referred to as "the knowledge acquisition bottleneck". Initial resources required were an English-Chinese machine-readable dictionary (MRD), a Chinese monolingual corpus and Chinese-English machine translation software. These were used to acquire an initial set English sense examples, which were then classified using machine-learning approaches. Ng et al. [Ng2003] also acquired English sense examples from English-Chinese parallel corpora. They grouped senses which shared the same Chinese translation, and thus the words on the English side of the corpus were considered to have been disambiguated and tagged by their Chinese translations. Their follow-up work successfully scaled up this approach [Chan2005] to perform well on the Senseval-2 English all-word task. However, a major problem with using parallel corpora is that not all senses of a word will necessarily be found in a corpus of finite length.

Yarowsky [Yarowsky1995] describes a mechanism for "bootstrapping" a sense tagger, where one begins with a small set of "seed" examples of two senses of a word. Those seed examples can then be augmented incrementally with additional examples of each sense. We can do this for *plant*, which has two senses: *plant life* and *manufacturing plant* - although in fact the method also works for words with three or more senses. Keywords in context are found for a fixed window size on either side for all instances of *plant* in the corpus. A small number of seed examples are chosen for each sense, and tagged (e.g. sense A for the sequence *used to strain microscopic plant life from the* and sense B for *automated manufacturing plant in Fremont*). Then, using the one sense per collocation principle, all the other training examples containing those collocates of plant are tagged - i.e. every context containing the word *life* is tagged as sense A, and every context containing *manufacturing*, is tagged as sense B. The remaining examples (typically 85-98%) remain as untagged residuals.

A decision list algorithm then examines all the tagged contexts, and identifies other collocates of *plant* that reliably partition the seed training data, ranked according to reliability as measured by the log likelihood (LL) ratio. These decision lists, containing rules such as *animal within 2 to 10 words suggests sense A with LL of 6.27,* are used to tag as many of the residuals as possible. At each iteration, new decision rules are learnt from already tagged examples, enabling more of the residuals to be tagged, and in turn more extensive decision lists can be created. In the final decision list, the two most reliable rules were as follows: if *growth* is the word immediately to the right, choose sense A with LL = 10.12; if *car* occurs within plus or minus k words, choose sense B with LL = 9.68. An evaluation of simple accuracy was 90.6% in this case using only two collocates as seeds.

Cong Li and Hang Li [Li2004] proposed a new method for WSD machine learning technique called "Bilingual Bootstrapping" (BB). BB makes use of, in learning, a small number of classified data and a large number of unclassified data in the source and the target languages in translation. It constructs classifiers in the two languages in parallel and repeatedly boosts the performances of the classifiers by further classifying data in each of the two languages and by exchanging between the two languages information regarding the classified data.

BB constructs classifiers for English to Chinese translation disambiguation by repeating the following two steps: (1) constructing classifiers for each of the languages on the basis of the classified data in both languages, (2) using the constructed classifiers in each of the languages to classify some unclassified data and adding them to the classified training data set of the language. The reason for classifying data in both languages in step (1) is that words in one language generally have translations in the other and their translation relationship can be found by using a dictionary. Experimental results indicated that BB significantly outperforms the existing Monolingual bootstrapping technique in word translation disambiguation.

4.2.2.6 Graph-based methods (unsupervised)

Most unsupervised WSD systems have employed the vector space model, where sense is represented by a vector of features, but graph-based algorithms have become more popular, since there are a number of ways graph connectivity may be exploited in unsupervised WSD systems [Navigli2007] [Agirre2006a]. In general, graph-based WSD algorithms consist of two stages: first a graph is built representing all possible interpretations of the word sequence being disambiguated - graph vertices correspond to word senses, and edges represent dependencies between senses, such as synonymy or antonymy. Secondly, the graph structure is assessed to determine the relevance of each vertex, and sense disambiguation amounts to finding the most important vertex for each word. The simplest way to measure the relevance of a vertex is by its in-degree, i.e., the number of edges terminating in it. A vertex is said to be central if it has high in-degree. While in-degree centrality gives a simple count of the number of connections terminating at a vertex, eigenvector centrality acknowledges that not all connections are equal. It assigns relative scores to all vertices in the graph, assuming that connections to vertices having a high score contribute more to the score of the vertex in question. PageRank [Page1999] and HITS (Hypertext Induced Topic Selection) [Kleinberg1999] are popular variants of eigenvector centrality that have been used in WSD. PageRank determines the relevance of a node *v* recursively, according to a Markov chain model. All vertices *u* that link to *v* contribute towards determining its relevance. Each contribution is given by the PageRank value of the respective vertex, *PR(u),* divided by the number of its neighbours or out-degree. HITS determines two values for each vertex *v*, the authority *a(v)* and the hub value *h(v)*. Intuitively, a good hub is a vertex that points to many good authorities, whereas a good authority is a vertex that is pointed to by many good hubs. A major difference between HITS and PageRank is that the former is computed on a sub-graph of relevant pages, whereas the latter takes the entire graph structure into account. According to the key player problem (KPP) algorithm, a vertex is considered important if it is relatively close to all other vertices, as determined by the sum of the inverses of the shortest path distances between *v* and all other vertices. Betweenness is based on the idea that a vertex is important if it is involves in a large number of paths, compared to the total number of paths. The betweenness of a vertex *v* is calculated as the fraction of shortest paths between pairs of vertices that pass through *v*.

4.2.3 Combination of knowledge sources

The process of WSD is illustrated by Rayson and Wilson's SEMSTAT [Thomas1996], a semantic tagger which reads in a text and assigns a code number standing for a particular word sense to each word in that text. For each word, a lexicon is first checked to see what senses that word can take.

Many words are unambiguous, but if more that one sense is possible for a given word, WSD techniques come into play, making use of the following types of information:

1. The part of speech (POS) tag assigned by the CLAWS POS tagger [Garside1987].
2. The general likelihood of a word taking a particular meaning, as found in certain frequency dictionaries.
3. Idiom lists are kept. If an entire idiomatic phrase is found in the text being analysed, it is assumed that the idiomatic meaning of each word in the phrase is more likely than individual interpretations of the words.
4. The domain of discourse can be an indicator. For example, if the topic of discussion is footwear, then "boot" is unlikely to refer to the boot of a car. This accords with Yarowsky's principle of "one sense per discourse" [Yarowsky1995].
5. Special rules have been developed for the auxiliary verbs *be* and *have*.
6. Use of collocation filters.

A number of systems use machine learning techniques for finding the **combination** of features (e.g., collocates, POS of words in the window) most likely to reveal word sense, notably Stevenson and Wilks [Stevenson2001]. They used a suite of techniques to filter out incorrect senses of words. They used part-of-speech filtering, degree of dictionary definition overlap, and collocation-based filtering.

They also used selectional preferences in the sense described by Wilks [Wilks1973]. LDOCE word senses are labelled with selectional restrictions expressed by 36 semantic codes. Named entities such as the names of persons and organisations can also be mapped onto these codes (human and abstract respectively). Relations between nearby words such as adjective-noun, subject-verb and verb-object are identified, and all senses of each word in the relation are considered in turn. According to the semantic codes of the word senses, restrictions are placed on the sentence, and combinations of word senses which do not fulfil these restrictions are filtered out. In the sentence "John ran the hilly course", *hilly* has only one word sense (undulating terrain) while *course* has two LDOCE senses: *route* and *programme of study*. The only word sense of *hilly* comes with the semantic restriction that it must modify a nonmovable solid. The route sense of *course* has the restriction that it must be of type nonmovable solid, which is consistent with the semantic restriction on *hilly*. However, *course* in the sense of a *programme of study* is restricted in that it must be of type *abstract*, which cannot be modified by *hilly*, so this second interpretation is rejected.

Use is also made of the LDOCE codes in the context of 50 words on either side of the ambiguous word, using an algorithm developed by Yarowsky [Yarowsky1995]. For each possible sense of each word in the window, the following quantity is maximised:

$$\sum_{w \in context} \log \frac{\Pr(w \mid SCat)}{\Pr(w)}$$

where Pr(w | SCat) is how likely it is to be word *w* given that its sense is *SCat.* The results from the above filters are combined using a machine learning algorithm called TiMBL memory-based learning, originally developed by Daelemans et al. [Daelemans2000].

4.2.4 Evaluation

The SENSEVAL project performs systematic annual evaluations of WSD algorithms.

It is an open evaluation exercise, taking place in 1998 with WSD tasks for English, French and Italian [Kilgarriff2000]. A corpus manually annotated with the correct sense of each word was used as a "gold standard", against which the output of each of the programs was compared. The SENSEVAL systems could make use of the rich HECTOR dictionary rather than a simple lexicon. In HECTOR, for each homograph, there is a separate entry for each sense distinction, including fields for word sense definition, POS information and examples of usage. Nowadays the WordNet hierarchy will is used rather than the HECTOR dictionary.

The **gold standard** or ground-truth corpus used for the training and evaluation of word sense disambiguation systems at Senseval is the Semcor 1.6 corpus distributed with WordNet [Miller1990], a thesaurus created at Princeton University. Semcor is a manually sense-tagged subset of the Brown corpus of over 200,000 words. Another gold standard is the TWA sense-tagged dataset [Mihalcea2003], which uses text drawn from the British National Corpus, sense-tagged for six words with two-way ambiguities: *bass, crane, motion, palm, plant*, and *tank.* Whichever gold standard is chosen, we incur the additional task of mapping our own dictionary senses to the set used by the gold standard.

The theoretical **upper** bound of performance of an algorithm is usually human performance - if human judges cannot agree on a task, we cannot expect an automatic procedure to do any better. Similarly, we can use human performance as a guide as to whether our disambiguation algorithms are given enough information to perform their task. For example, we can design an experiment to find how well can humans can disambiguate using just three words of context on either side. If the humans can't do it, then more information, such as a wider context, must also be provided for a machine to do it. The lower bound or baseline is the performance of the simplest possible algorithm, which in the case of SENSEVAL is to assume that the most common sense of a word is always the correct one.

A metric commonly used to evaluate WSD performance is **accuracy**, which is simply the percentage of automatically sense-tagged words in a text which have been assigned the correct sense tag. Accuracy is an intrinsic measure, showing how good WSD performance is *per se*. Extrinsic measures of WSD performance show to what extent WSD is useful for some greater task, such as retrieval of web pages from a search engine.

Sanderson [Sanderson2000] in pseudo-word experiments concluded that improvements in search engine effectiveness with WSD would result only if the disambiguation was at least 50-60% accurate (the interpolated break-even point). Reasons search engines can work well without any WSD at all are the skewed distribution of word senses and collocation query effects. Stokoe [Stokoe2005] also aimed to find the minimum disambiguation accuracy required to improve search engine performance, and also the level of granularity with which one must define word sense for this to happen. With homonymy, two senses of a given word such as "bat" ("flittermouse" or "cudgel") are distinct, with no underlying relationship between these meanings which have come about independently from differing root languages. With polysemy, both meanings subsumed by a higher concept, for example the "mouth" of the human face and the "mouth" of a river are both openings. Coarse grained disambiguation assigns all nouns to one of 25 top-level Wordnet categories. Sanderson [Sanderson2000] felt that that fine-grained may not offer benefits over coarse-grained, which can be done with greater accuracy, but Stokoe found that fine-grained distinctions can help. Using pseudowords, Stokoe simulated disambiguation to varying degrees of accuracy and measured the impact this had on search engine effectiveness. In cases where additional homonymy had been added to the corpus, disambiguation accuracy at or above 76% was required for disambiguation to be of benefit. However, the breakeven point is only 55% where additional polysemy has been added. This showed that retrieval effectiveness is more sensitive to polysemy than homonymy.

Carpuat and Wu [Carpuat2005] evaluated WSD algorithms by their effect on the quality of the statistical machine translation of Chinese sentences into English, as measured by the BLEU metric which quantifies the number of matching n-grams (sequences of adjacent characters between 2 and 4 words long) between the machine output and a "gold standard" translation produced by human translators. In one approach they used their WSD sense predictions to constrain the set of English sense candidates considered by the MT system for each of the target words. In a second approach, the WSD predictions were used to post-process the output of the MT system: if different, the MT output of the target word was directly replaced by the WSD prediction. In each case, they concluded that even state-of-the-art supervised WSD does not help the BLEU score. Reasons for this may have been that their MT model was already sufficiently accurate, and/or the n-gram based model was unable to produce a better score even when the WSD approach is able to suggest a better target word.

4.2.5 Scalability of WSD Approaches

In this section we will consider a number of scalability issues affecting WSD. Supervised approaches need vast amounts of sense-tagged data, and gathering hand-labelled data is time consuming and costly. This is the so-called knowledge acquisition bottleneck. Other issues are that the VITALAS project needs WSD which works for unrestricted text, not specific narrow topics, and which enables fine-grained sense distinctions to be made.

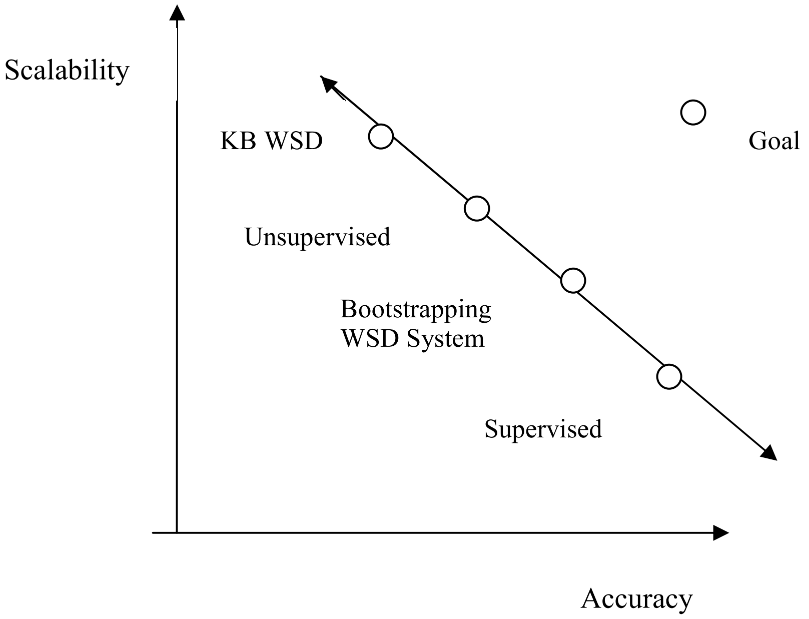


Figure 12: Scalability versus Accuracy of WSD Systems. From McInnes (2007).

Figure 12 shows the trade-off between scalability and accuracy in current WSD systems. At one end of the continuum supervised WSD systems give the best accuracy, but the knowledge acquisition bottleneck is that large amounts of manually sense-tagged training data is needed to provide enough examples to learn every possible word that we might want to disambiguate (giving broad coverage), making them intractable for large-scale problems such as information retrieval [McInnes2007]. At the other end of the spectrum we have what McInnes refers to as knowledge-based WSD systems. These include some of the approaches covered in this report, as follows:

- using an ontology such as WordNet to assign a score as to how similar or related two words are to each other.
- contextual knowledge-based, where statistical measures such as mutual information and Kullback-Liebler divergence are used to infer collocations between words rather than hand-label them individually.
- frequency based, such as Mihalcea and Moldovan's [Mihalcea1999] method of counting the frequency of word pairs to determine their senses.
- Graph-based, as discussed in section 2.7.

These methods increase the scalability of WSD approaches, since they are automatic approaches which can be applied cheaply to large quantities of raw text, but do not achieve the same accuracy as hand-labelled training data. The goal of VITALAS is to achieve high scalability without sacrificing accuracy. McInnes [McInnes2007], working in the biomedical domain, suggests that the step forward is to make use of the UMLS, a large knowledge-base of medical concepts. Since VITALAS wishes to create WSD approaches which are domain independent, we must make full use of the WordNet hierarchy of general English, and the ontologies provided by our partners who maintain large archives. Not shown in Figure 12 are the old **Artificial Intelligence**-based approaches, which have poor scalability since a set of rules specific to every sense of every word had to be hand crafted. These rules enforced constraint satisfaction, where for example the word "grasp" would mean to grasp something physically rather than intellectually if the following rules were satisfied: a) the object of the sentence is a physical object, and b) the action is one of physical contact [Shann1984].

Deibel [Deibel2004] tries to identify robust techniques that circumvent the need for massive amounts of training data. One solution is to use large publically-available sets of training data, one of the most popular being the **Semantic Concordance.** Based on the Brown corpus, a million-word collection of tagged texts using in natural language programs, the Semantic Concordance consists of over a hundred texts in which the words have been hand-annotated according to their WordNet senses. The Semantic Concordance may be accessed on http://www.cs.unt.edu/rada/downloads.html

Mihalcea and Moldovan [Mihalcea1999] use the **internet** as an unlimited source of training data containing examples from all genres and domains. They used the hit count of word pairs on the Intenet to determine the most likely senses for each word, with an accuracy of over 80%. Their technique was to take a verb-noun (V N) pair, and choose one of them, such as the noun. They then find the synsets of each possible sense of the noun. For the m words in each synset, the following query is input to a search engine: (V N1) OR (V N2) OR … OR (V Nm). The number of hits found by each sense of N reflects the likelihood of that sense being the true one.

Chan and Hwee [Chan2005] obtain large amounts of training data by gathering examples from **parallel corpora**, and found that this approach is scalable to a large set of nouns. The basic idea is that all cases of an English word translated by the same Chinese word must take the same sense. They found for some data sets including SEMCOR, classifiers trained on parallel text examples outperformed classifiers trained on manually sense-tagged data. As discussed in section 2.6, the expense of building a parallel corpus might be overcome by using machine translation to translate monolingual corpora. Other techniques described in this report to overcome the need or vast amounts of sense-tagged data, are artificially ambiguous pseudowords (see Section 2.5), and bootstrapping approaches which start with very small amounts of training data from which more training data can be inferred in a recursive process (see Section 2.6).

Another issue of scalability is that **coarse-grained** WSD (where senses correspond to general categories) is much easier for robust, broad-coverage applications than **fine-grained** WSD. Ciaramita and Altun [Ciaramita2006] report that many fine-grained sense distinctions are too subtle to be captured automatically, and wide range of classes (the "class space") means the problem is not tractable to scalable machine learning methods. In addition, there is generally insufficient sense-tagged data for training such a model. Ciaramita and Altun [Ciaramita2006] suggest a two-stage process where semantic tagging according to coarse-grained sense distinctions is performed first, to cut down the class space at the later fine-grained disambiguation stage. In particular, they suggest pre-processing with a probabilistic semantic tagger based on a Hidden Markov Model (HMM), which stores information about likely sequences of coarse-grained sense distinctions in text. In this respect it would work analogously to a probabilistic syntactic part-of-speech tagger such as CLAWS [Garside1987]. Thus the WordNet broad categories, of which there are only 41 in total (26 for nouns and 15 for verbs, examples being *artefact, shape, stage, animal, plant* and *action*) provide a partial disambiguation step which helps by reducing the number of possible senses for each word [Ciaramita2006]. POS tagging is a largely solved problem, where state-of-the-art taggers achieve near-perfect accuracy. Syntactic tagging is easier than semantic tagging, as there are a relatively small number of parts-of-speech, but the semantics of words occupy a much larger space of possibilities [Deibel2004]. This again suggests the wisdom of prior coarse-grained word sense tagging before attempting fine-grained WSD. A similar approach is advocated by Kohomban and Lee [Kohomban2005], where coarse-grained distinctions are learnt first, then these general concepts are transformed to fine-grained word senses using simple techniques such as collocational knowledge and part-of-speech tagging.

Stokoe et al [Stokoe2003] simulated a large-scale scenario in which WSD was used to improve search engine performance. Rather than attempting to perform WSD for every single word in the vocabulary of a large document collection, they concentrated on providing high quality WSD for only those words which appeared in the set of test queries.

EuroWordNet is a multilingual semantic lexicon for several European languages, including English, French and German, and is structured similarly to WordNet. Each language specific (Euro)WordNet is linked to all others through an inter-lingual index, which is based on WordNet 1.5. Via this index the languages are interconnected, so that it is possible to move from a word in one language to similar words in any of the other languages. EuroWordNet is a very efficient WSD resource when applied to large scale corpora. EuroWordNet has a very high coverage of terms and provides a detailed network of lexical relations between them. We will recommend the use of EuroWordNet in the VITALAS project, and aim to disambiguate words by mapping them on to fine-grained EuroWordNet senses. EuroWordNet can also help with the process of WSD itself, making use of the relations between terms and the comprehensive set of dictionary definitions for each word sense.

4.3 Conclusions

A baseline text indexing module in the VITALAS system should satisfy at least the following functional and technical requirements: open index structure to allow faceted search and example based searches, robustness and ability to process large volumes of documents. As the vectorial model is the simplest one answering these requirements and as it is used in most search engines nowadays, we suggest implementing it as a state of the art indexing component in the first version of VITALAS system. To enable scalability this component should support distributed indexing and search mechanism. One possibility would be to use the Hadoop architecture (http://lucene.apache.org/hadoop/) that is an open source implementation of the MapReduce framework and that should provide a strong basis for implementing a robust distributed search and indexing engine.

For more advanced versions of VITALAS text search engine, we would recommend the use of the multi-lingual semantic lexicon EuroWordNet as a suitable lexical resource for large-scale WSD problems.

5 Similarity search scalability: index structures and access methods

5.1 Introduction

One of the major objectives of the VITALAS project is to enable the use of very large documents databases, up to several millions of still images and several ten thousand hours of video materials. This large datasets are for the moment rarely handled in the literature. As a comparison, the corpus of the most popular video benchmark TRECVID grew from 11 hours in 2001 to 158 hours in 2006.

Enabling the scalability of search engines requires specific indexing structures and search algorithms. Whereas pure text search engines are now mature enough to deal with huge datasets, large scale content-based multimedia search engines are still facing some challenging issues. Content-based search engines retrieve, compare or classify multimedia documents via the use of compressed representation of their contents. As these representations are not simple textual or scalar values, their management requires the efficient resolution of *similarity queries* on less or more complex objects to be compared using arbitrary metric or non-metric similarity measures.

Similarity queries generally consist either of searching in a database the similar objects of a given query object, known as nearest neighbour search, or of determining the similarity relation between two sets of objects, called the similarity join operation. In the second case, often a similarity self-join is of interest, in essence computing a (usually partial) similarity matrix between all objects in the database. The naïve method to solve similarity queries is to compare the objects exhaustively, one by one. Obviously, this technique leads to poor performance, with linear search complexity for nearest neighbour queries and a quadratic complexity for similarity joins.

For the last 10 years, strong links have been initiated between computer vision and database communities leading to more or less efficient strategies, depending on the type, dimension or complexity of the handled objects and similarity measures. Most of these strategies avoid scanning sequentially the whole database by structuring, compressing or transforming the feature space in a pre-processing step and then having specific search algorithms.

In the following, we start to discuss some general concerns in section 5.2, including a brief review of classical index structures on which a lot of similarity search techniques are based. Section 5.3 deals with a synthetic review and analysis of similarity search methods whereas section 5.4 is dedicated to conclusions regarding VITALAS project.

5.2 Preliminaries

5.2.1 Types of queries

Historically, the most frequently used type of query in similarity search is the K-nearest neighbour query (KNN). Its goal is to identify the k nearest neighbours of a query object according to a given similarity measure, usually a metric distance (a metric that satisfies the triangle inequality).

In many applications, similarity search is mapped to nearest neighbour search in a high dimensional space. Data and queries are transformed to high dimensional points and the most similar data instance to a query is defined by the nearest neighbour of the query point. Example applications that adopted this paradigm include image and multimedia databases, time-series databases, and data mining. In the former, the visual characteristics of images (e.g., colour distribution, texture patterns, image structure) are represented by feature vectors of high dimensionality (e.g., colour histograms alone vary from 64 to several hundreds of bins). The similarity between two images is then defined by the distance of the corresponding feature vectors in the high dimensional space. Time series analysis attempts to identify similarity patterns and trends in time-evolving data. The similarity between two time-series is often defined by their distance in a metric space. The third application, data mining, aims at analysis of large collections of data records. These data are represented by high dimensional points, after mapping the attributes to dimensions. Nearest neighbour search is also applied here to find the most similar records to a query instance, e.g., in order to identify clusters (i.e., dense regions in the high dimensional space) that reveal interesting correlations between attributes.

Range queries are related to k-NN queries, but request all the neighbours under a given threshold of similarity. Processing range queries has however been studied to a lesser degree.

5.2.2 Types of data and similarity measures

It is possible to categorize the similarity search methods according to the type of features and the type of similarity measure they can handle:

1. **VSLp (Vector Space and Lp norm)** = methods working only in vector spaces with Lp distances as similarity measures. The features being compared can only be vectors of same dimensionality. If p is specified, the class is more restrictive: VSL1, for example, correspond to the methods working only with L1 distance.

2. **VS (Vector Space)** = methods working only in vector spaces but with any distance measure.

3. **MS (Metric Space)** = methods working on any set of objects and any metric distance measure.

4. **NMS (Non Metric Space)** = methods working on any set of objects and any similarity measure, even non metric ones.

5.2.3 Index structures

Traditional methods for accelerated access to relational data are not readily applicable for similarity queries on high dimensional or complex data. Therefore, there has been a continuous development of specific index structures, which aim to extend conventional indexes in order to manage data of higher dimensionality.

An index structure organizes the data in order to facilitate and to accelerate the access in a storage structure (the database). This index structure is composed of data that points to the features in the storage structure. The storage structure is usually organized in pages; each page provides access to a group of descriptors. The storage structure can reside on disk or in main memory. Most of the time, during the search process, the index structure is stored in the main memory even in the case of storage structures on disk, in order to accelerate the search process and to limit disk access.

5.2.3.1 One-dimensional index structures

In this section, we briefly discuss the two most popular approaches used for one-dimensional data indexing, since multidimensional indexing structures are often extensions of the same underlying principles and because a lot of similarity search methods are directly based on one-dimensional index structures as background technology.

5.2.3.1.1 Linear indexes

This type of index structure is the simplest one used for one-dimensional data. The data or the descriptors are sorted according to a quantized one-dimensional value associated to each feature. The quantized value can be obtained by different techniques such as hash functions, space filling curves or projections.

The index structure is a simple array where each cell contains a pointer to the data having the same quantized values.

This index structure is very efficient in terms of computational costs and very simple to develop. Nevertheless, it has two main drawbacks. The dynamic insertion of a new feature is not possible, since it requires to sort again the whole set of quantized values. Moreover, there are many cells that are empty. Multi-level indexes (extendible hashing, dynamic hashing) or sequences of hash functions (linear hashing) have been proposed to address this problem.

5.2.3.1.2 B-trees

B-Trees [Comer1979] and its variants (B+-tree, B+ tree, B*-tree, B#-tree) are tree-based index structures for one-dimensional data. The concept of this structure is to group the data into intervals. At each level of the tree, the intervals have the same number of elements. B-Trees are then balanced structures. B-trees are m-ary structures that allow to make dynamic insertions, deletions and searches in $O(\log_m(n))$.

The B-tree is an efficient structure for one-dimensional data that has been developed with advanced algorithms for handling concurrent updates. Because of its maturity, it is the most popular index structure for the implementation of relational databases and file systems.

5.2.3.2 Multidimensional index structures (VS and VSLp)

A lot of effort has been spent to extend the classical one-dimensional index structures to multidimensional vectors. A high number of structures were proposed in the literature with the main objective to solve the curse of dimensionality. Complete surveys of such methods already exist, e.g. in [Gaede1998] and [Bohm2001] or more recently in [Samet2006].

As the B-tree does, most multidimensional index structures hierarchically partition the data according to their proximity in the feature space. Vectors are stored in the leaves of a tree whose nodes correspond to a region of the multidimensional space. Three categories of methods are distinguished depending on the nature of the partitioning: space-partitioning index structures, data-partitioning index structures or space filling curves.

5.2.3.2.1 Space partitioning index structures

In this kind of structures, the data are grouped according to joint and non-overlapping regions of the multidimensional space. The KD-tree [Bently1975] is the most popular technique of this family. It is a binary tree dividing the regions of the feature space in two other regions, at each level of the tree, only along one dimension (or along one hyper-plane in more advanced versions). It guarantees complete partitioning without overlap. As the KD-tree is not balanced, variants have been proposed to solve this problem (Adaptive KD-tree [Bently1979], the KDB-tree [Robinson1981], LSD-tree, LSDh-tree [Henrich1996]).

The main advantage of space-partitioning structures is that they are simple to create and to handle. Without overlap, the insertion of a point and the split of the regions are easily supported operations. The drawback is that the whole space is structured, even where there is no data. During a search, this will yield useless accesses.

The pyramid technique [Berchtold1998], was proposed to achieve efficient range queries within very high dimensional data spaces. The pyramid technique divides the data space in 2d pyramids (linear partitioning) which tops are converging into the centre of the feature space. A data point may thus be approximated by the pyramid it belongs to and its distance to the pyramid's top. Hence, using this mapping, the database can be indexed using a B+-Tree and the points may be accessed in O(log(n)). A range query is performed by accessing all pyramid slices intersecting with the query hyper-sphere and evaluating each data points within these slices. However, the pyramid technique fails to provide efficient queries when many slices are touched by the query. This may be the case when the database is not uniformly distributed. To address this issue, the authors have proposed the extended pyramid technique which uses as the top of the pyramid the mean data point of the database. However, this approach is only valid for unimodal distribution of the database.

5.2.3.2.2 Data partitioning index structures

In data partitioning index structures, the data are grouped according to possibly disjoint or overlapping regions of the multidimensional space. The regions corresponding to the nodes of the tree may overlap and the union of all regions does not systematically cover all the feature space. A vector can thus belong to several regions of the same level of the tree. The advantage is that it indexes only the regions containing vectors. The drawback is that several sub-trees may need to be accessed for one single position in the feature space.

Most popular technique is the R-tree [Guttman1984] which can be thought of as a multidimensional generalization of B+-tree. It is a balanced tree that models database partitions as Minimum Bounding Rectangles (MBR). A minimum bounding rectangle is the smallest hyper-rectangle that encloses a set of feature vectors. The regions are constructed by splitting full nodes of the tree in order to conserve the balancing. A lot of R-Tree variants have been proposed [Manolopoulos2003]. The main difference between these methods is the splitting strategies, which change the amount of overlap between the MBR. The different strategies modify the selectivity of the access method. The most efficient R-Tree is the R*-Tree [Beckmann1990], thanks to its better selectivity.

The SS-Tree (Similarity Search Tree) [White1996] uses Minimum Bounding Spheres (MBS) instead of MBRs. The advantage of using spheres is that they more compact than MBR. Therefore, the search process is accelerated because the filtering rules are more discriminative, and the selectivity is higher. However, splitting MBS involves a large amount of overlap and is more time consuming.

The SR-Tree (Sphere-Rectangle Tree) [Katayama1997][Bouteldja2006a] is designed to take advantage of using both MBRs and MBSs. The SR-Tree has been evaluated to be more efficient than the R*-Tree and the SS-Tree [Li1999].

The X-tree (eXtended node Tree) [Berchtold1996] method is a cost-based extension of the R-tree structure for data spaces with high dimensions. In fact, an X-Tree is an adaptable access method which is reduced to an R*-Tree when overlaps are tolerated and can be a sequential scan if there is a unique super node (containing all points) when no overlap is tolerated. The X-Tree is one of the most efficient multi-dimensional access methods which is suited for data with medium dimensionality.

5.2.3.2.3 Space filling curves

Multi-dimensional vectors can be projected into one-dimensional space filling curves. The advantage is that these projections can be handled with a one dimensional index structure such as a B-Tree. Space-filling curves have the advantage to preserve locality, which means that two points that are close on the curve remain close in the feature space. The reciprocal rule is unfortunately not true and exact similarity queries require a lot of interval queries in the one dimensional index. Furthermore, determining these intervals can also be a difficult task. Most used space filling curves are Z-ordering, Peano and the Hilbert curves [Lawder2000a, Lawder2001, Lawder2000b].

5.2.3.3 Metric index structures (MS)

Metric index structures are based on data partitioning but contrary to multidimensional index structures, the structuring is only based on the distance between the data and is agnostic to the data representation. This kind of techniques is very useful when the objects of the database cannot be represented by vectors although a metric exists to compare them (e.g. when the objects are graphs, trajectories or strings). This is also useful when using complex metrics that have no direct meaning in the Euclidean space. A complete list of metric data structures can be found in [Chavez2001] and [Samet2006]. Most of them are based on pivots [Moore2000] [Yianilos1993] [Ciaccia1997]. The principle is to use some objects of the database, called pivots, as references to index the position of the other objects by their distance to these pivots. Search is then usually performed by using the triangle inequality properties. Among these techniques, the M-Tree (Metric Tree) [Ciaccia1997] is the most popular and efficient metric access method that has been proposed. The data points are grouped around a set of representatives. At each level of the M-Tree a set of m data points is chosen and the data points are associated to their closest representative. An internal node stores the representatives along with their covering radius, i.e. the distance from the representative to its farthest points. When used on vectors, the M-Tree has been demonstrated to be more efficient than an R*-Tree. This can be explained by the fact that the M-Tree behaves similar to an SS-Tree, since the feature space is structured according to balls around pivots.

5.2.4 Curse of Dimensionality

In the context of optimisation, R. Bellman in [Bellman1961] was the first to observe that partitioning the solution space is not efficient for high-dimensional spaces, since the number of partitions increases exponentially with the dimensionality. The effect has become known as the ‘curse of dimensionality’.

Increasing the number of dimensions involves other undesired phenomena, see also [Beyer1999, Weber1998]. Among the effects most critical for nearest neighbour search applications is that in high-dimensions, the distances between all pairs of vectors converge, such that all points become equi-distant. Obviously, if this happens, identifying the nearest neighbours becomes irrelevant (since the nearest neighbour of a query is at the same distance as the farthest [Beyer1999]).

The curse of dimensionality affects most of the common indexing structures and search methods, causing serious performance degradation already at about 10 dimensions, and breaking down on most data sets for dimensionalities higher than 15 to 20. As we will see in the next section most of the similarity search methods try to limit the effects of the dimensionality, through space transformations, index projections or query approximations, but the curse of dimensionality remains a fundamental problem for access methods.

5.3 Similarity search methods

A complete and up-to-date survey of efficient similarity search methods can be found in [Zezula2007]. In the following, we focus mainly on techniques that have been successfully applied to large multimedia content indexing and retrieval purposes.

5.3.1 Exact nearest neighbours queries in multidimensional and metric index structures

Historically, the first solution developed to speed up similarity queries was to perform exact nearest neighbour queries in multidimensional or metric index structures. Such queries are usually based on geometrical filtering rules to eliminate directly some branches of the trees (i.e. regions of the feature space) without accessing the objects contained in their leaves. Most of these rules are based on the concept of MinDist and MaxDist: given a query Q and a region R of the feature space, MinDist is the minimal distance between the query Q and any point of R, and MaxDist is the maximal distance between the query Q and any point of R. Most popular filtering rules are then :

If MaxDist(Q,R1) ≤ MinDist(Q,R2) then the branch corresponding to R2 can be pruned.

If MinDist(Q,R) ≤ d_k , where d_k is the maximal distance of the current best solution, then the branch corresponding to R can be pruned.

Different strategies have been then developed to improve these filtering rules according to the specific shapes of the index structure (e.g. the HS algorithm for the R-tree [Hjaltason1995]) or to improve the course of the tree (e.g. the RKV algorithm [Roussopoulos1995]).

The main problem of these techniques is that they strongly suffer from the dimensionality curse. The number of visited leaves grows exponentially with the dimension, and the performance of these techniques become worse than a simple sequential scan when the dimension is higher than about 15 [Weber1998].

5.3.2 Feature space transformation

5.3.2.1.1 Dimension reduction techniques (VS)

One of the first solutions to overcome the curse of dimensionality phenomenon was to use dimension reduction techniques such as KLT, SVD or DFT (see [Gerbrands1981, Chakrabarti2000]) coupled with a multidimensional structure and exact nearest neighbours queries. These methods exploit the underlying correlation of vectors and/or their self similarity [Korn2001], frequent in real datasets.

The problem of these techniques is that they do not preserve the metric of the initial space. Therefore, the neighbours of the query are found in the transformed space and not in the initial space. Thus, dimension reduction techniques introduce uncontrolled inaccuracies in the NN-search results. Moreover, such techniques are effective only if the transformed space dimension becomes small enough which depends on the initial dimension and on the distribution of data in the initial feature space.

Dimension reduction techniques are thus often used to pre-process the data, but are not sufficient to accelerate search in most datasets.

5.3.2.1.2 Space transformation based on metric (MS)

Another way to embed the data in a lower dimensional vectorial space, is to use metric based space transformation techniques [Bourgain1985] [Faloutsos1995] which are only based on the distances between the objects and thus agnostic to the objects representation. Such techniques are generally used as an alternative to dimension reduction techniques for non-vector data or complex similarity measures. A multidimensional index structure coupled with nearest neighbours queries can then be applied to perform similarity queries [Cheung2003, Faloutsos1996, Hjaltason2003]. Such techniques usually suffer from the same problems than dimension reduction techniques, i.e. inappropriate dimension of the transformed space and uncontrolled inaccuracies in the results.

5.3.3 Spatial approximations based similarity search structures

Geometric approaches enforcing approximate NN-searches have originally been studied (on low dimensional data) in the computational geometry community [Bentley1980, Arya1993, Bern1993]. The concept has then been applied to high dimensional data in order to limit the effects of the curse of dimensionality. Such approaches typically consider an approximation of the sizes of cells instead of considering their exact sizes. They account for an additional value ε when computing the minimum and maximum distances to cells, making somehow cells smaller [Arya1993]. Shrunk cells tend to make the geometric filtering rules more severe, which, in turn, decrease the number of visited cells. Cells containing interesting vectors might be filtered out, however. The approximate nearest neighbours search strategy for M-Trees presented in [Ciaccia2000] also relies on a single value ε set by the user. Here, ε represents the maximum relative error allowed between the distance from the query and its exact NN and the distance from the query and its approximate NN. In this scheme, ε can only have a very obscure meaning to the user and its setting is far from being intuitive. In addition, their experiments showed that, in general, the actual relative error is always much smaller than ε. Ciaccia and Patella [Ciaccia2000] also present an extension of AC-NN called PAC-NN which uses a probabilistic technique to determine an estimation of the distance between the query and its NN. It stops the search as soon as it founds a vector closer than this estimated distance. PAC-NN assumes however a known data distribution around the query, which might not be possible in the general case. The main advantage of geometric query approximation techniques is that they can be performed in any generic index structure, allowing easy indexing of all kinds of data, balancing of the pages and fast dynamic insertions. Most of them have a sub-linear search complexity in database size for moderate dimensions. The main drawback of geometric query approximation techniques is that the benefits of the approximation on the search time are far from optimal. Acceptable inaccuracies in the results may allow only insignificant accelerations. As said before, the settings of the approximation parameters are not intuitive. Furthermore, the performance of geometric query approximation techniques still degrades strongly when the dimension becomes high (> 50).

Ferhatosmanoglu et al. [Ferhatosmanoglu2001] have proposed a clustering-based approximation algorithm for vector spaces that has achieved good results for moderate-to-high dimensions. As a pre-processing step, their algorithm partitions the data into a large number of small clusters using a variant of the well-known K-means heuristic (here, K refers to the number of clusters sought). To handle a query, the clusters are ranked according to their similarity with the query vector. A tentative set of approximate NNs is generated by sorting the elements of the closest clusters according to their distance from the query, based only on the first few coordinates of the vector representation. The tentative solution is then iteratively refined by considering more clusters and/or more vector coordinates. This technique can potentially speed up query processing by an order of magnitude or more. Despite these substantial improvements, their method has several drawbacks. First, the authors offer no clear way of determining when the search should be terminated, an issue that limits application of the method in practice. Also, the quality of the query result depends heavily on the success of clustering. Although K-means is popular due to its efficiency, it is notorious for producing poor-quality clustering, particularly due to its sensitivity to the initial choice of representative vectors available to it. Their method also requires that upper and lower bounds be imposed on the cluster sizes, which can result in a cycle of cluster creation and destruction that would prevent the K-means heuristic from terminating. Li Chang et al. [Li2002] proposed a different clustering-based scheme called Clindex, which makes use of a fast index structure over a set of clusters formed via a bottom-up cluster-growing technique, relative to a grid partition of the data domain. In experiments with a 48-dimensional set of 30,000 image vectors, Clindex achieved speedups of roughly 21 times over sequential search at the 70% recall level, and roughly 12 times at the 90% recall level. Clindex is designed to operate with the Euclidean distance as the similarity measure, and requires the data to be well partitionable into clusters. Its main drawbacks are the computational cost of indexing and the difficulty of determining when the search should be terminated, an issue that limits application.

Another way to introduce spatial approximations is to include geometric tolerance directly in the index structure. The spill-tree (sp-tree) [Liu2004], for example, is a variant of metric-trees in which the children of a node can "spill over" onto each other, and contain shared data points. Unlike standard metric trees, the children of a spill tree node can share objects. This strategy results in a larger index but allows speeding up the search algorithm since some backtracking can be avoided. The accuracy of the results is however difficult to control and the size of the index can become prohibitive for high dimensional data.

In [Houle2005], Houle et al. introduce a practical index for approximate similarity queries of large multi-dimensional data sets: the spatial approximation sample hierarchy (SASH). A SASH is a multi-level structure of random samples, recursively constructed by building a SASH on a large randomly selected sample of data objects, and then connecting each remaining object to several of their approximate nearest neighbours from within the sample. Queries are processed by first locating approximate neighbours within the sample, and then using the pre-established connections to discover neighbours within the remainder of the dataset. The technique can return a large proportion of the true neighbours roughly two orders of magnitude faster than sequential scan for moderate dimensions, and less than one order of magnitude for extremely high dimensions (up to 1 million dimensions). Main drawbacks are the very high computational cost of the pre-processing step and the absence of query accuracy control.

5.3.4 Probabilistic similarity search

Probabilistic similarity search techniques have been proposed to optimise the benefits of approximate nearest neighbour searches. The idea is to replace the common geometric filtering rules by probabilistic filtering rules which select only the regions of the feature space having the highest probability to contain a correct answer. This allows reducing the number of visited cells comparing to geometric rules, as the latter only take care of the presence or absence of a geometrical intersection. Furthermore, thanks to the underlying probability model, it is possible to control the quality of the search more accurately.

5.3.4.1 Clustering based probabilistic similarity search techniques

These techniques cluster the dataset during the indexing step. The descriptors of each cluster are stored together, where the index structure consists of the list of the clusters (instead of a tree, as in common data partitioning index structures).

Bennet et al in [Bennett1999] proposed the DBIN (Density Based Indexing) method. This approach is based on the estimation of the distribution in the database. This distribution is modelled by a Gaussian mixture estimated with the Expectation Minimization (EM) algorithm. During the search process, the clusters are selected according to the probability that they contain a nearest neighbour of the query. This probability is estimated by a normal law. A cluster is not visited if his probability is lower than a given threshold. The advantage of DBIN is the possibility to control very accurately the probability of missing nearest neighbours. However, the EM algorithm has a very high computational costs increasing with the number of clusters. This technique is thus limited to small sets of large clusters, which are then however insufficiently selective to acquire efficient access for very large datasets.

Berrani et al [Berrani2002] propose a related method, based on a large number of small clusters. The clustering is based on a BIRCH improved algorithm [Zhang1996]. During the indexing step, for each cluster, an optimal radius is estimated according to error bounds. The approximate search strategy uses this estimated radius to control the rate of missed nearest neighbours.

These techniques can be very efficient on relatively small datasets and offer an accurate and intuitive control of the quality of nearest neighbour queries. They however have some major drawbacks:

- The clustering step requires a high computational cost and remains problematic for very large datasets. It also prevents dynamic updates of the index.
- The number of clusters is a crucial parameter that may affect the technique's performance drastically. Tuning according to the size of the dataset and its distribution is however difficult.

5.3.4.2 Distortion based probabilistic similarity search techniques

In these techniques, the probabilistic model does not model the distribution of the features in the dataset, but rather the distribution of neighbours relevant for a given query. Whereas clustering based techniques are probabilistic versions of K nearest neighbour queries, distortion-based techniques are more related to range queries, since the model is agnostic to the dataset content.

In [Joly2004], Joly et al. defined distortion-based probabilistic queries relying on the distribution of the relevant similar features for finding a transformed image or video. Finding that the probability of a tolerated transformation decreases when the "amplitude" of the transformation increases, Joly et al. proposed to model the effect of tolerated transformations on a signature by an isotropic multidimensional Gaussian probability density function, and to perform probabilistic retrieval based on this model transformed document. They use a space partition index structure based on the Hilbert space filling curve. Probabilistic retrieval then consists in selecting a minimum number of cells such that their cumulated probability (following the model) is above a fixed threshold. They show in [Joly2007] that the search cost of the technique is sub-linear in database size up to a given size and report some real-world experiments on very large video databases including several tens thousands hours of video (more than 1 billion 20-dimensional features).

In [Poullot2007], Poullot et al. proposed some improvements of the technique (ZPSS, for Z-curve Probabilistic Similarity Search). To speed up the computation of the keys (cell addresses) they used a Z-grid that appeared to be more efficient than the Hilbert curve. To improve the balancing between the populations of the cells, they also defined an adaptive version of the Z-grid taking into account the distribution of the dataset along each component. The probabilistic retrieval is also improved by ranking the component-wise exploration of the feature space according to decreasing uniformity of the components. This allows improving significantly the hierarchical pruning of the cells. An experiment is performed on a huge database including more than 5 billion 20-dimensional features.

Distortion-based probabilistic techniques have several advantages:

- Searching most probable signatures is more efficient than using geometric criteria
- As the probabilistic modelling of the queries is supposed to be independent of the database distribution, the selection of the visited domain regions can be determined without access to the database. This makes the method highly efficient (no tree is needed) and easily applied in a distributed setting.
- The indexing computational cost is low since it only consists of ranking the data according to their key on the space-filling curve.
- Distortion-based probabilistic techniques were successfully applied to huge datasets including several billions of signatures and have experimentally proved to be sub-linear in database size.

Main drawbacks of these techniques are:

- The difficulty to perform dynamic updates of the index
- The unknown behaviour for very high dimensional data.
- The difficulty to define relevant but efficient probabilistic models for the queries.

5.3.5 Projections based similarity search techniques

In these techniques, the dimensionality problem is solved by working in index spaces of lower dimensionality, thanks to projections in the feature space. Contrary to space transformations techniques, the features of the dataset as well as the computed distances remain unchanged; the transformed spaces are only used to determine which pages need to be accessed. In [Kleinberg1997], J. Kleinberg showed that an algorithm projecting the database on m random lines is able to perform an epsilon-NN query with an error bound epsilon related to m. The aim is to have m lower than the dimension of the descriptors. These projections reduce the influence of the dimensionality curse. Several techniques are based on this principle.

5.3.5.1 LSH and variants

Indyk [Gionis1999, Indyk2000] used and developed this concept by proposing the LSH method (Locality Sensitive Hashing). This method projects the vectors into the Hamming cubes defined by hash functions and then uses several hash functions (hash tables) in order to drive the precision of searches or the probability to retrieve an element. This technique became very popular in the last few years thanks to its very good performances on very large datasets and its robustness to increasing dimensionality. The first version of LSH was however dedicated only to the L1 metric. More recent works extended the approach for Lp distances (LSH p-stable [Datar2004]) by defining new family of hash functions; associated search cost models were provided recently [Andoni2006].

The projections used in the first version of LSH were chosen arbitrarily. Therefore, the index structure is not tuned according to the data contained in the database. [Yang2004b] (Hierarchical-Non-Uniform LSH or HNULSH) has proposed to adapt the hash functions according to the statistics of projections.

LSH is based on multiple sets of hash functions that involve multiple indexes. LSH methods may require over a hundred hash sets to perform accurate search. In order to reduce this number of hash functions, Panigrahy [Panigrahy2006] proposed a Point-Perturbation based LSH approach (PPLSH) to reduce the space requirement of the original LSH approach. The approach generates some randomly "perturbed" objects in the neighbourhood of the query object. The perturbed objects are all used as queries and the final result set is the union of all results. Based on the concept of perturbations, Hash-Perturbation LSH (HPLSH) has been proposed in [Lv2006]. Instead of applying perturbations on the query object, hash values are perturbed. The experimentation in [Lv2006] shows that this perturbation approach allows dividing by 5 the number of hash tables for the same accuracy of search.

The main advantages of LSH approaches are their efficiency for large sets of high dimensional vectors and the possibility to insert dynamically new vectors. Another advantage is the possibility to design new hash functions adapted to specific metrics or data.

The main drawback of LSH approaches is the difficulty to control the accuracy of the search. The control of the parameters remains obscure. Another important drawback is the full size of the index structure since it can be composed of hundreds of one-dimensional hash tables.Finally, the complexity of the search time when the database size increases is not known since it was not applied to more than 3 millions feature vectors.

5.3.5.2 MedRank

A second family of approaches based on projections combine random projection with another rank aggregation [Fagin2003, Lejsek2005]. The method of aggregation (e.g., MedRank [Fagin2003]) replaces the original complex distances in order to reduce computational cost. [Lejsek2005] presents the OmedRank approach, which is an implementation of MedRank using an efficient index structure called the PvS Index; an experiment is presented in [Lejsek2006]. During the indexing step, the OmedRank method randomly selects d lines (d being lower than the dimension of the feature space). Each vector of the database is projected onto the d lines and the d projected values are used as indexes in d independent B+-trees. The leaves of the B+-trees only contain identifiers of the feature vectors and not the vectors themselves in order to limit the storage extra-cost. During the search step, a query vector q is projected on the d lines and the projected values are searched in the d B+-trees in a round robin fashion. During these scans, the MedRank algorithm counts how often each descriptor identifier is encountered. If a particular descriptor identifier has been met more than ½ d times, it is returned as a nearest neighbour.

The main advantages of OmedRank approach are:

- The strong reduction of the number of computed distances (thanks to the MedRank algorithm that replaces standard nearest neighbour search algorithm).

- the use of B+-trees allowing to insert and delete dynamically a descriptor in the index structure,
- its validation on very large databases (more than 200 million of 24 dimension descriptors)

The main drawbacks of OmedRank are:

- the difficulty to control the accuracy of the search
- the size of the index structure (at least d B+-tree indexes)
- the final score of the retrieved elements is not the original distance to the query

5.3.6 Sequential techniques

The efficiency of nearest neighbour queries in multidimensional or metric index structures is known to degrade fast with dimensionality (curse of dimensionality) and, as a result, they can be outperformed by sequential scan when dimensionality is high (e.g., for 64 dimensions or more). Reading the whole dataset in order to evaluate a nearest neighbour query is sometimes faster than using a partitioning index. Some search techniques therefore try to benefit from the efficiency of sequential scans while limiting the computational cost of each comparison.

5.3.6.1 Vector approximation techniques

In the VA-File technique [Weber1998], a sequential scan is performed on vectors that approximate the original ones. The feature space is quantized in 2.d.b cells (d being the dimension of the space and b the number of bits per dimensions) in such a way that cells are nearly equally populated by data points. The VA-File stores the approximation of every data point, in the order of the original data file. To perform the query, a multi-step algorithm is applied which uses a lower and a upper bound of the distance between a point and a cell. As the storage size of the approximations is much lower than the original data size, scanning the VA-File is much more efficient (higher page capacity). However, exact data points in the data file have still to be accessed in the second step of the algorithm. Hence, even if the scan of the VA-File, which is linear with the size of the database, is faster than the sequential scan, the overhead implied by the random access to the data-file may make the approach less efficient, depending on the selectivity of the first step. In particular for highly clustered database, the bound on distance are not enough precise and the selectivity of the first step degenerates. To solve this issue, more bits may be used to quantize the data points. But that way, the introduced overhead will decrease the page capacity of the VA-File. Hence, Tuncel et al. [Ferhatosmanoglu2000] have proposed the VA+-File. The idea is to perform the quantization not directly on the data space but on a transformation of this space that is more adapted to the distribution of data points. In their work, the authors use a PCA. Hence, the VA+-Files has a better selectivity than VA-Files for non-uniformly distributed databases. In [Yianilos1993], the authors proposed a different approximation scheme. They add more information to better locate data points. Instead of increasing the number of bits r, they enrich the approximation by the polar coordinates of the data points within their associated cell. That way the LPC-Files (Local Polar Coordinate) have a better selectivity than a VA-File, for a given number of bit used for the approximation.

In [Weber2000], Weber and Böhm also presented an approximate nearest neighbours query strategy for the VA-File called VA-BND. As standard geometrically approximated queries (see section 5.3.3), a tolerance ε is empirically estimated by sampling database vectors. They show that this parameter is big enough to increase the filtering power of the rules while small enough in the majority of cases to avoid missing the true nearest-neighbours. The main drawback of this approach is that the same ε is applied to all existing cells. This does not account for the possible very different data distributions in cells, making this scheme efficient but not reliable in terms of precision.

The main advantage of vector approximation techniques is that they can accelerate searches in very high dimensional spaces more efficiently than using index structures. The computational cost of the indexing is very low and linear in database size. Dynamic updates are also very easy.

The main drawback is that the complexity of a search is linear in database size since the gain against sequential scan is constant. Thus processing huge datasets remains problematic. Furthermore, these techniques are usually profitable only for a disk storage of the data. When all the vectors can be stored in memory, the gain against classical sequential scan becomes too small or sometimes they even work worst.

5.3.6.2 Search on vertically decomposed data

In [DeVries2002], an unconventional physical design alternative is used, that maintains a separate table for each dimension, containing, of all vectors in the repository, the coefficients of that same dimension. This physical representation accommodates a novel search technique called Branch-and-bound ON Decomposed data (BOND). In BOND, the distance between the query point and all data vectors is accumulated by scanning these dimensional projections one-by-one. After processing few of them, partial distances of each vector to the query are known; then, lower and upper bounds on the complete distance of the k-nearest neighbours are exploited to discard safely from further consideration those vectors that cannot possibly participate in the response set. Applying this process iteratively reduces the candidate set such that the last stages are performed on just a small database sample.

As BOND is not based on a static space decomposition, where all dimensions are of equal importance, it enables efficient evaluation of weighted k-NN queries, where dimensions can have different (arbitrary) importance at search, and queries on arbitrary dimensional sub-spaces.

The advantages of BOND are summarized as follows:

- It avoids a large number of computations compared to a full sequential scan which guarantees better performance.
- It is conceptually simple, causes practically no storage overhead, and requires no pre-processing of the data. Dynamic updates are simple, and its performance improvement has shown to be orthogonal to that of compression (like in the VA-File).
- Its good performance is robust to increasing dimensionality (assuming the selectivity in each dimension can be predicted from the query).
- On the same data representation different variants of k-NN queries can be processed efficiently, including queries with different weights of importance on the various dimensions, queries in dimensional subspaces, queries with various similarity metrics or multi-feature queries that combine similarities from various sources.

The main drawback is that the search cost of the first step of BOND is linear in database size whatever the data. Thus processing huge datasets may remain problematic – although indexing each dimension individually could address this potentially, the resulting structure is more complicated, so warrents further experimental investigation before conclusions can be drawn.

5.3.7 Batch similarity search and similarity join

In a lot of multimedia retrieval applications, a query document is represented by a set of feature vectors (e.g. visual local features in an image), where an independent nearest neighbour search is first performed for each feature vector. Completing all the NN searches, an overall similarity is then computed. As nearby feature vectors in a document may be similar, processing them together may reduce the number of accessed cells and computed distances.

Braunmuller et al. [Braunmuller2000] were the first to study multiple similarity queries for mining in metric databases. They propose a general framework for answering simultaneously a set of similarity queries (k-NN and sphere queries) and give algorithms both for reducing I/O cost and CPU time. An experiment is provided for multiple k-NN queries that evaluates the speed-up for a linear scan of the collection of points as well as for an X-tree traversal. It is shown that parallelization yields an additional significant speed up. They propose some CPU saving by using the triangle inequality and two lemmas. For the X-tree, the saving is said to be of the order of 2 for two databases. In [Bouteldja2006b], Bouteldja et al. propose and evaluate several algorithms for multiple sphere queries in a collection of points indexed by a tree structure. Their work uses similar ideas to Braunmuller et al. for reducing I/O cost and CPU time, but is applied to sphere queries instead of k-NN queries. For improving CPU time they use the distance triangle inequality relying on the lemmas proposed by Braunmuller as well as three other novel lemmas. Last, Braunmuller et al. propose incremental processing of a multiple feature query to allow providing partial answers to the user at an early stage of the evaluation. The algorithm performs the usual single depth first traversal of the tree. Each relevant node is accessed only once in both algorithms. An extensive evaluation of the impact of applying the triangle inequality on the saving of expensive distance computations, does not lead to the same conclusions as Braunmuller, since the saving is about one order of magnitude in most cases. This difference shows the high impact of the queries distribution. Such techniques are indeed efficient only on highly redundant batched queries.

Another batch nearest neighbour search strategy more dedicated to video processing is proposed by Shao et al. [Shao2007]. The algorithm is based on dynamic query ordering (DQO). The idea is that the overlapped candidates (or search space) of a previous query may help to further reduce the candidate sets of succeeding queries. DQO aims to progressively find a query order such that the common candidates among queries are fully utilized to maximally reduce the total number of candidates. Modelling the candidate set relationship by a Candidate Overlapping Graph (COG), DQO iteratively selects the next query to be executed based on its estimated pruning power to the rest of queries with the dynamically updated COG. The experiments the significance of this strategy for low dimensional data (<16) but higher dimensions lead to mitigate savings of the order of 2.

Batch similarity search can provide benefits in a different manner from searching simultaneously redundant queries of a single document. In [Joly2007] and [Poullot2007], a batch strategy is used to avoid random disk accesses in very large datasets that do not fit in main memory. They accumulate a large amount of queries depending on the available RAM, on the size of the database and on disk latency, and process all these queries on parts of the database that are successively loaded in memory. The underlying idea is similar to the partitioning schemes common in traditional relational join processing, e.g., the partitioned hash join. This strategy is particularly efficient thanks to the grid-based index structures used in these schemes allowing fast computation of the keys without any access to the index structure. In-memory performance can be maintained for databases up to 100 times larger than the main memory capacity (In [Poullot2007], a successful experiment is performed on a dataset of 9 billion 20-dimensional feature vectors (250 Gb)). Furthermore, as the provided saving is independent from the query distribution, the accumulated queries may come from very different sources: parallel search of different documents, multiple users, etc.

In spirit, the goal of batch similarity search is somewhat similar to similarity join processing [Zezula2003], which finds the nearest neighbours for each point in one set from another set (or the nearest neighbours for each point in one set from itself). However, similarity join usually deals with two large datasets that cannot fit into main memory, while the query set of batch similarity search strategies is usually smaller so that all the query points can remain memory-resident. Recently, some efficient nearest neighbour join algorithms have been proposed. The algorithm by Böhm [Bohm2004], termed MuX uses the DFS algorithm to compute the neighbourhoods of one block at a time (i.e., it computes the nearest neighbours of all points in the block before proceeding to compute the nearest neighbours in other blocks) by maintaining and updating a best set of neighbours for each point in the block as the algorithm progresses. The rationale is that this will minimize disk accesses as the nearest neighbours of points in the same block are likely to be in the same set of blocks. The GORDER method [Xia2004] takes a slightly different approach in that although it was originally designed for high-dimensional data-points, it can also be applied to low-dimensional datasets. In particular, this algorithm first performs a principal component analysis (PCA) to determine the first few dominant directions in the data space and then all of the objects are projected to this dimensionally reduced space, thereby resulting in drastic reduction in the dimensionality of the point data set. The resulting blocks are organized using a regular grid, and, at this point, a nearest neighbour algorithm is performed which is really a sequential search of the blocks. Even though both the GORDER and the MuX methods compute the neighbourhood of all points in a block before proceeding to process points in another block, each point in the block keeps track of its nearest neighbours encountered thus far. Thus this work is performed independently and in isolation by each point with no reuse of neighbours of one point as neighbours of a point in spatial proximity. Instead, in [Sankaranarayanan2007], the authors identify a region in space that contains all of the nearest neighbours of a collection of points (the space is termed locality). Once the best possible locality is built, each point searches only the locality for the correct set of k nearest neighbours. This results in large savings. Also, this method makes no assumption about the size of the data set or the sampling-rate of the data. Experiments of this technique show that it is faster than both the GORDER and the MuX methods and performs substantially fewer distance computations.

5.3.8 Distributed similarity search structures

Centralized similarity search structures achieve a significant speedup when compared to the sequential scan. However, experience with centralized methods reveals that query execution cost increases in most cases linearly with the growth of the dataset when it becomes very huge. Furthermore, the limitation of the main memory size of one single computer usually causes strong degradations of the performance beyond a certain size of the dataset (the exact size threshold depending on the technique). In this section, we present some recent methods which try to solve these problems by exploiting parallel computing power. The idea is easy in principle: as the dataset grows in size, more independent computation and storage resources are added (CPUs, memory, disks, etc.), keeping the query response time low. One possibility is to randomly partition the data, building a separate index structure for each partition. However, at query time, this would require each query be run through all the index structures. While this could be done in parallel, the overall query throughput would be limited. Another alternative is to make a more intelligent partition of the data in order to centralize high level filtering steps and to distribute independent refinement steps. Not all similarity search structures can be efficiently distributed since it requires reducing as much as possible data-dependent parameters and computations during the centralized filtering step.

Bawa et al. propose the LSH forest [Bawa2005], a distributed similarity search structure based on the well-known technique of locality-sensitive hashing (LSH, see section 5.3.5.1). It improves upon previous designs by eliminating the different data-dependent parameters for which LSH must be constantly hand-tuned, and by improving on LSH's performance guarantees for skewed data distributions while retaining the same storage and query overhead. They show how to construct this index in main memory, on disk, in parallel systems, and in peer-to-peer systems. Unfortunately, it was only applied on text corpora and not on more complex content-based features.

In [Novak2006], Zezula et al. propose the M-Chord technique, a distributed data structure for metric-based similarity search. The structure takes advantage of the idea of a vector index method iDistance in order to transform the issue of similarity searching into the problem of interval search in one dimension. Whereas the standard version of iDistance processes similarity search as multiple interval queries in a single B+-tree, the M-chord structure is composed of m independent distributed B+-trees. Each B+-tree contains only the data having the same hash according to a hash function that divides the mono-dimensional iDistance index in m load-balanced intervals. At query time, a centralized step determines each B+-tree must be accessed and in which interval. The proposed peer-to-peer organization, based on the Chord protocol, distributes the storage space and parallelizes the execution of similarity queries. The drawback of this technique is that it is mainly dedicated to exact range queries and nearest neighbour searches remain costly. The influence of the dimension and the size of the dataset are also not clearly studied.

In [Batko2006], the authors propose a distributed access structure which is fully dynamic and exploits a Grid infrastructure. The technique is based on the D-index structure, which provides a hashing-like centralized similarity index. The D-index structure is a pivot-based metric index structure but is not a tree. It uses pivot-based distances as hash functions that split the data into separable buckets. A specific mapping is then used to assign the buckets of the D-index structure to the nodes of the grid. This translation of bucket identifications to node identifications allows having more buckets stored on the same node. It also permits to replicate buckets, i.e., the same bucket can be kept on multiple nodes. When a user poses a similarity query, the master node identifies the buckets that might contain relevant data. The bucket-to-node mapping is consulted to get addresses of Grid nodes. The master node initiates the execution phase by forwarding the query to the respective Grid nodes where the objects qualifying the query get retrieved. The individual sub-answers are then sent back to the master node which merges them to form the final answer to the query. The drawback of this technique is that it is mainly dedicated to exact range queries and nearest neighbour require a lot of hand tuning and remain costly. The influence of the dimension and the size of the dataset are also not clearly studied.

In [Liu2007], the authors describe a distributed nearest neighbours search structure based on the spill-tree technique ([Liu2004], see section 5.3.3). They first create a random sample of the data small enough to fit on a single machine and build a metric tree for this data. Each of the leaf nodes in this top tree then defines a partition, for which a spill tree can be build on a separate machine. At query time, to avoid costly backtracking over the different machines, each query object is speculatively sent to multiple leaf sub-trees when the query appears to be too close to the boundary. This is effectively a run-time version of the overlap buffer usually used in spill-trees at tree building time. This technique has been successfully applied to the clustering of a huge image descriptors database, including about one billion 100-dimensional features. The overlap buffer width of the spill-tree is however a critical parameter that is not easy to tune while having a strong influence on the performances. In the reported experiments, the overlap buffer width can be very small since the clusters of the dataset contain only highly similar objects (near-duplicate images). The real accuracy of the search regarding the true nearest neighbours is unfortunately not given.

5.3.9 Comparisons and evaluations

Unfortunately, there is today no benchmark initiative or common evaluation corpus to compare similarity search structures. Some discussions are currently in progress trying to solve this lack but it is important to notice that such comparisons are difficult due to the wide variety of approaches and the different aspects to be evaluated. The search cost itself, which is probably the most important, can be compared only on the same data and at constant number of neighbours, constant accuracy of the search, identical machines, etc.

To give a more comprehensive survey of the reviewed similarity search techniques, we summarize in Table 2 some quantitative and qualitative comparison criteria. We only treat the techniques that have been recently evaluated on large datasets of high dimensional content-based features:

- **Section:** reference to the section of this document in which the technique is presented.
- **Dimension**: range of dimensions on which the technique was successfully evaluated.
- **Storage**: type of data storage for which the technique was successfully evaluated (memory / disk / both)
- **Static/Dynamic**: ability of the technique to perform efficient dynamic insertions and deletions.

- **Accurate quality control:** precise if the technique allows an accurate control of the search quality.

- **Index storage:** Qualitative cost of the storage space requirement of the structure (low / medium / high).

- **Indexing cost**: low / medium / high / very high

- **Large scale experiment**: we give some indicative performances on the largest found experimented dataset for each technique. Note that these costs are not really comparable since usually the data are not the same as well as the machines and the accuracy of the search which is rarely given with the same criterion. However this gives a good idea of the scalability of the technique.

Technique	**Section**	**Dimension**	**Storage**	**Static / Dynamic**	**Accurate quality control**	**Index storage**	**Indexing time**	**Large scale experiment**
SASH [Houle2005]	5.3.3	[30-1 Million]	Memory	Static	No	High	Very high	DIM=32, N=9 millions Search time = 28 ms (one CPU)
Spill-tree [Liu2004] [Liu2007]	5.3.3 5.3.8	[60-3800]	Memory	Dynamic	No	High	High	Dim=100, 1.5 billion Search time = 48 ms (distributed on 2000 CPU)
Clustering based technique [Berrani2002]	5.3.4.1	[20-512]	Disk	Static	Yes	Low	High	Dim=24, N=5 millions Search time = 500 ms (one CPU)
ZPSS [Joly2007] [Poullot2007]	5.3.4.2	[10-32]	Memory	Static	No	Low	Low	Dim=20, N=16 billions Search time = 40 ms (batch mode, one CPU)
LSH [Indyk2000] [Datar2004]	5.3.5.1	[20-500]	Memory	Dynamic	No	High	Low	Dim=192, N=2.6 millions Search time = 200 ms (one CPU)

HPLSH [Lv2006]	5.3.5.1	[20-500]	Memory	Dynamic	No	Medium	Low	Dim=192, N=2.6 millions Search time = 200 ms (one CPU)
OmedRank [Fagin2003] [Lejsek2005]	**Error! Reference source not found.**	[10-120]	Disk	Dynamic	No	Medium	Low	Dim=24, N=20 millions Search time = 25 ms (one CPU)
BOND [DeVries2002]	5.3.6.2	[26-260]	Both	Dynamic	Yes	Low	Low	Dim=166, N=60,000 Search time = 20 ms (one CPU)

Table 2 Comparison of some similarity search structures

5.4 Conclusions

5.4.1 Back to VITALAS data

Efficient similarity search structures will be required in the VITALAS system for low level features extracted from audio and/or visual data, and also for cross-media or semantic vector indexes. There exists no structure covering all the usages of VITALAS system and we will thus discuss technological choices with a finer granularity according to different types of datasets and search strategies. In this section, we will define some typical feature datasets in order to discuss the requirements of VITALAS in terms of similarity search structures.

Typical size of the dataset foreseen for the end of the project is **10,000 hours of video** (including sound tracks) and **3,000,000 images**. In the following, we refer these two datasets as **VDB** and **IDB**.

Some VITALAS functionalities such as learning, browsing, visualization or refinement searches will only work on smaller subsets of documents (program category, photo-reportage, learning datasets, etc.). To discuss the requirements for such datasets, we define two theoretical subsets **VDB-SMALL** and **IDB-SMALL** respectively containing **15 hours of video** and **10,000 images.** Notice that some working datasets could even be smaller than that (one single movie of 2 hours, small learning datasets of 500 images, etc.).

From **IDB** dataset we will typically generate two kinds of features databases depending on the nature of the extracted features (global or local):

- **IDB-GLOBAL: 3 million 500-dimensional** feature vectors (about 6 Gb). Typically composed of 5 histograms of 100 dimensions extracted in each image if **IDB**. Histograms could represent global visual attributes or visual bag of words. Such a dataset could also represent vectorial semantic or cross-media features.

- **IDB-LOCAL**: **3 billion 30 dimensional** feature vectors (about 100 Gb). Typically composed of 1000 local descriptors extracted in each image of **IDB**.

Same kind of features extracted in **IDB-SMALL** will typically give the following datasets:

- **IDB-SMALL-GLOBAL**: **10,000 500-dimensional** feature vectors (about 20 Mb).

- **IDB-SMALL-LOCAL**: **10 millions of 30 dimensional** feature vectors (about 300 Mb).

From **VDB** we will typically generate two kinds of feature datasets:

- **VDB-GLOBAL: 72 million 500-dimensional** feature vectors (about 150 Gb). Typically composed of high dimensional visual, audio or semantic features extracted every half second.

- **VDB-LOCAL**: **18 billion 30 dimensional** feature vectors (about 600 Gb). Typically composed of 500 local features per second of video.

Same kind of features extracted in **VDB-SMALL** will typically give the following datasets:

- **VDB-SMALL-GLOBAL**: **110,000 500-dimensional** feature vectors (about 250 Mb).

- **VDB-SMALL-LOCAL**: **27 million 30 dimensional** feature vectors (about 900 Mb).

5.4.2 VITALAS requirements

5.4.2.1 Dimensionality

As seen in the previous section, the VITALAS system will typically deal with either medium dimensions (between 10 and 50) or high dimensions (between 100 and 1000). Techniques performing only with low dimensions are thus to be avoided (typically exact searches in multidimensional indexing structures) and there is no need for techniques allowing the use of very high dimensions (typically higher than 10,000) such as SASH for example (see section 5.3.3).

5.4.2.2 Storage

Regarding the predefined typical feature datasets, we can draft the following conclusions:

- Small feature databases can always fit in main memory (**IDB-SMALL-GLOBAL**, **IDB-SMALL-LOCAL**, **VDB-SMALL-GLOBAL** and **VDB-SMALL-LOCAL**). The maximum size is obtained for local features extracted in 15 hours of video and is less than 1 Gb which is far from recent memory sizes (up to 64 Gb). It is thus possible to manage them with a fully in-memory similarity search structure.

- High dimensional features computed on a large image dataset (**IDB-GLOBAL**) can nowadays also fit in the main memory of one single state-of-the-art machine. It is thus also possible to index them with a fully in memory similarity search structure.

- High dimensional video features (**VDB-GLOBAL**) cannot be entirely stored in the main memory of one single machine. Some functionality of VITALAS system will however require only off-line or delayed similarity queries which could be processed by an in-memory batch strategy or similarity join (see section 5.3.7). For real-time similarity search queries, such a dataset will require an efficient disk storage similarity search structure or a distributed in-memory architecture (see section 5.3.8).

- Almost same conclusions are applicable to image local features from IDB (**IDB-LOCAL**) since the generated amount of data is almost the same than **VDB-GLOBAL.** The difference, however, is that a disk strategy is not affordable with standard hard disks. When using local features, the number of queries per document is indeed multiplied by several orders of magnitude**.** The response time requirements are thus stronger and even a very efficient disk strategy with few accesses per query would give poor performances on standard disks.

- Large local video features datasets **(VDB-LOCAL)** require the most scalability efforts and can thus also not be entirely stored in the main memory of one single machine. In-memory batch strategies or similarity joins are possible for off-line processes or delayed response time requirements. For real-time similarity search queries however, a distributed in-memory architecture will be required since disks are nowaday still too slow. To limit the number of machines, a good load balancing will be an important requirement for the design of the distributed technique.

5.4.2.3 Dynamicity and indexing cost

Fully dynamic index structures allowing the efficient insertion or deletion of one single feature in the dataset are not essential for VITALAS. At indexing time, VITALAS documents will indeed never be handled one by one.

However, the indexing process must be very fast for small datasets (**IDB-SMALL-GLOBAL**, **IDB-SMALL-LOCAL**, **VDB-SMALL-GLOBAL** and **VDB-SMALL-LOCAL**) and must remain low for very large datasets (**IDB-GLOBAL**, **IDB-LOCAL**, **VDB-GLOBAL** and **VDB-LOCAL**):

- Small subsets of data may indeed be constructed in-line by a user and will have to be searchable as fast as possible. Efficient bulk loading strategies should thus be preferred to multiple individual insertions requiring a lot of distance computations and comparisons (e.g. metric trees).

- Very large datasets will off-course be indexed off-line but with so large volumes of data, a too high algorithmic complexity of the indexing process is not affordable. In practice, the indexing complexity must not exceed O(NlogN): a complexity O(N^2) e.g., with a simple very fast operation (10 ns), would require 100,000 years to index **VDB-LOCAL**.

5.4.2.4 Structure size

The additional storage cost of the index structure is also an important criterion. Several recent efficient methods use multiple index structures (e.g. LSH [Indyk2000], OmedRank [Lejsek2005]), or add complex links between data (e.g. SASH [Houle2005]) or even replicate some of the features (e.g. spill-tree [Liu2004]), in order to ensure the search efficiency or/and the accuracy. As the additional storage cost usually depends on parameter, it is difficult to estimate it definitively for each technique. However, we can draft some general conclusions:

- Search structures having an excessive additional storage cost should be avoided for high dimensional large datasets (**IDB-GLOBAL, VDB-GLOBAL**). To limit the global storage cost of VITALAS system, the additional storage cost should not exceed the size of the data itself for in-memory structures.

- Disks being cheaper, the additional storage cost can be larger for disk storage structures, up to 10 times the initial data size for standard disks.

5.4.2.5 Search accuracy

The search accuracy requirements depend on the dataset type and on the targeted applications:

- Small high-dimensional features datasets (**IDB-SMALL-GLOBAL**, **VDB-SMALL-GLOBAL**) will require a very good accuracy of the search and a sufficiently good control of it. With such datasets, the similarity measure is indeed directly used to compare, rank or cluster the documents. For instance, performing a K-means clustering with 5% of false nearest neighbours will lead to errors of the same order of magnitude in the final clusters. Small high dimensional features datasets will also be used in VITALAS for machine learning based services (e.g. learning step of object recognition or relevance feedback). In these techniques, even a very far neighbour can have a high impact on the final classifier and approximate searches must be used very carefully.

- Large high-dimensional features datasets (**IDB-GLOBAL**, **VDB-GLOBAL**) also require a good accuracy and a sufficiently good control but approximate search techniques will be more profitable. The relevant nearest neighbours in a very large dataset are indeed closer to the query than in a small one. Spatial or probabilistic approximations can thus prune large regions of the feature space without degrading too much the accuracy of the results.

- Local features datasets (**IDB-LOCAL, VDB-LOCAL, IDB-SMALL-LOCAL¸ VDB-SMALL-LOCAL**) can tolerate stronger approximations and less control accuracy. Local features are indeed only partial descriptions of a document (or a part of it). For instance, an image can be well retrieved even if only a quarter of its local features have been retrieved. A fine control of the accuracy is also less important since the accuracy of each local query can be very different while keeping a relatively good average accuracy that is usually sufficient for most applications.

5.4.3 Recommendations

According to the previous analysis of the state-of-the-art and the above mentioned VITALAS requirements, we draft here some recommendations:

- A basic sequential scan (dealing with different standard metrics) should be integrated in VITALAS system as a baseline technology. It will first be useful to compare and evaluate the performances of new developed techniques. Secondly, it could be an efficient solution to manage small high-dimensional features datasets (**IDB-SMALL-GLOBAL**, **VDB-SMALL-GLOBAL**) for complex operations, such as clustering or learning.

- An efficient sequential technique, such as BOND or VA-file (see section 5.3.6) should also be integrated and experimented to manage accurately and efficiently high dimensional features datasets (**IDB-SMALL-GLOBAL**, **VDB-SMALL-GLOBAL, IDB-GLOBAL**, **VDB-SMALL-GLOBAL**). BOND is certainly more appropriated than VA-files since it works both in memory and on disk, it causes practically no storage overhead and requires no pre-processing of the data. Furthermore, the BOND framework allows to speed-up a wide range of complex similarity queries, including queries with different weights on the various dimensions, queries in dimensional subspaces, various similarity metrics or multi-feature queries that combine similarities from various sources. New dedicated similarity queries on cross-modal datasets could thus be easily experimented in the scope of VITALAS.

- Projection-based and distortion-based similarity search structures (see respectively section 5.3.5 and 5.3.4.2) seem to be the most promising techniques to efficiently manage very large datasets of features. They share, in fact, several common suitable technical features that make them suitable for most VITALAS requirements:

  - Their search cost is experimentally sub-linear in database size.

  - They have very low indexing costs.

  - They are easily distributable (the indexes are not data-dependent and thus enable easier search parallelization, in-memory similarity joins or batch strategies).

  - They have been successfully experimented on very large features datasets close to VITALAS ones and were up to 3-4 orders of magnitude faster than a basic sequential for in-memory storage.

  We thus recommend integration of one of these techniques as a state-of-the-art module in VITALAS system.

  Some experimental and research efforts are however still required to design a novel similarity search structure that can efficiently manage similarity search queries in the larger datasets of VITALAS (**VDB-LOCAL**, **VDB-GLOBAL**, **IDB-LOCAL**): The drawbacks of projection-based methods are the additional storage cost and the search accuracy that is very difficult to control and usually low for acceptable performances. The number of projections is also a parameter difficult to tune. Distortion-based techniques are in fact not so far from projection-based techniques since the used index structure can be seen as one single multidimensional projection whereas projection-based techniques use several one-dimensional projections. The use of one single multidimensional projection allows to reduce the additional storage cost and to enhance the selectivity and the accuracy of the search. On the other hand it does not allow the management of high dimensional data (>30). The use of several low dimensional projections could be a solution to benefit from the advantages of both approaches. It however raises some issues that will need to be solved: How many projections and which dimensionality? How to choose the better projections according to the dataset? How to control the accuracy of the search?

Two other promising research axes related to projection based structures could be explored within VITALAS: the definition of data replication indexing strategies to allow efficient similarity search on disk storage and the evaluation of Solid Sate Disks as new storage type.

- An efficient distributed in-memory architecture will have to be developed to allow fast on-line multi-users similarity queries in the very large datasets of VITALAS (**VDB-LOCAL**, **VDB-GLOBAL**, **IDB-LOCAL**). If we design an easily distributable similarity search structure such as the one previously discussed, main remaining issues to be solved are the design of an effective load balancing strategy and the development of efficient data exchanges.

6 References

[A4Vision2007] A4Vision, http://www.a4vision.com/, June 2007.

[Agirre2006a] E. Agirre, D. Martinez, O. de Lacalle and A. Soroa, Two Graph-Based Algorithms for State-of-the-Art WSD, in Proc. of the Conference on Empirical Methods in Natural Language Processing, 585-593, 2006.

[Agirre2006b] E. Agirre, O. de Lacalle and D. Martinez, Exploring Feature Set Combinations for WSD, in Proc. of Congresso de la SEPLN, 283-284, 2006.

[Aldous1985] D. Aldous, Exchangeability and related topics, in Ecole d'ete de Saint-Flour XIII, 1-198, Springer, 1985.

[Amores2005] J. Amores, N. Sebe and P. Radeva, Efficient Object-Class Recognition by Boosting Contextual Information, in Iberian Conf. on Pattern Recognition and Image Analysis, 28-35, 2005.

[Amores2006] J. Amores, N. Sebe, and P. Radeva, Fast spatial pattern discovery integrating boosting with constellations of contextual descriptors, in Proc. of IEEE CVPR, 2006.

[Andoni2006] A. Andoni and P. Indyk, Near-Optimal Hashing Algorithms for Near Neighbor Problem in High Dimensions, in Proc. of the Symposium on Foundations of Computer Science, 2006.

[Argaw2005] A. A. Argaw, L. Asker, R. Cöster, J. Karlgren and M. Sahlgren, Dictionary-Based Amharic-French Information Retrieval, in Proc. of CLEF 2005, 83-92, 2005.

[Arya1993] S. Arya and D. Mount, Approximate nearest neighbor queries in fixed dimensions, in Proc. of ACM/SIGACT-SIAM Symp. on Discrete Algorithms, 1993.

[Baeza1999] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, in ACM Press, 1999.

[Barras2006] C. Barras, X. Zhu, S. Meignier and J.L. Gauvain, Multistage speaker diarization of broadcast news, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1505-1512, 2006.

[Batko2006] M. Batko, V. Dohnal and P. Zezula, M-Grid: similarity searching in grid, in Proc. of Int. Work. on information Retrieval in Peer-To-Peer Networks, 2006.

[Baumberg2000] A. Baumberg, Reliable feature matching across widely separated views, in Proc. of IEEE CVPR, 774–781, 2000.

[Bawa2005] M. Bawa, T. Condie and P. Ganesan, LSH forest: self-tuning indexes for similarity search, in Proc. of Int. Conference on World Wide Web, 2005.

[Bay2006] H. Bay, T. Tuytelaars, and L. Van Gool, Surf: Speeded up robust features, in European Conference on Computer Vision, 2006.

[Bayer1972] R. Bayer, E. M. McCreight, Organization and Maintenance of Large Ordered Indices, in Acta Informatica, 1, 173-189, 1972.

[Beckmann1990] N. Beckmann, H-P. Kriegel, R. Schneider and B. Seeger, The R*-tree: An efficient and robust access method for points and rectangles, in Proc. of ACM SIGMOD Int. Conf. on Management of Data, 322-331, 1990.

[Belkin1993] N. J. Belkin, Interaction with Texts: Information Retrieval as Information-Seeking Behavior, in Information Retrieval, 55-66, 1993.

[Bellman1961] R. Bellman, Adaptive Control Processes, in IEEE Transactions on Circuits and Systems, 1961.

[Belongie2002] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts, in IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(4), 509–522, 2002.

[Bennett1999] K.P. Bennett, U. Fayyad and D. Geiger, Density-Based Indexing for Approximate Nearest-Neighbor Queries, in Knowledge Discovery and Data Mining, 1999.

[Bentley1980] J. L. Bentley, B. W. Weide and A. C. Yao, Optimal expected-time algorithms for closest point problem, in ACM Transactions on Mathematical Software, 6(4), 1980.

[Bently1975] J. L. Bently, Multidimensional binary search tree used for associative searching, in Communications of ACM, 18(9), 509-517, 1975.

[Bently1979] J. L. Bently and J. H. Friedman, Data structures for range searching, in ACM Computing Surveys, 11(4), 397-409, 1979.

[Berchtold1996] S. Berchtold, D. A. Keim and H-P. Kriegel, The X-tree: An Index Structure for High-Dimensional Data, in Proc. of Int. Conference on Very Large Databases, 1996.

[Berchtold1998] S. Berchtold, C. Böhm and H-P. Kriegel, The Pyramid-Technique: Towards Breaking the Curse of Dimensionality, in Proc. of ACM SIGMOD Conf., 142-153, 1998.

[Berliner1979] H. Berliner, The B* tree search algorithm: A best-first proof procedure, in ArtificiaI Intelligence, 12, 23-40, 1979.

[Bern1993] M. Bern, Approximate closest point queries in high dimensions, in Information Processing Letters, 45, 1993.

[Berrani2002] S-A. Berrani, L. Amsaleg and P. Gros, Approximate k-Nearest-Neighbor Searches: A New Algorithm with Probabilistic Control of the Precision, Technical Report N°4675, INRIA Report, 2002.

[Beyer1999] K. Beyer, J. Goldstein, R. Ramakrishnan and U. Shaft, When is “nearest neighbor” meaningful?, in Proc. of Int. Conf. on Database Theory, 217-235, 1999.

[Beyerlein2002] P. Beyerlein, X. Aubert, R. Haeb-Umbach, M. Harris, D. Klakow, A. Wendemuth, S. Molau, H. Ney, M. Pitz and A. Sixtus, Large vocabulary continuous speech recognition of Broadcast News - The Philips/RWTH approach, in Speech Communications, 37(1-2), 109-131, 2002.

[Biatov2006] K. Biatov and J. Köhler, Improvement Speaker Clustering Using Global Similarity Features, in Proc. of Int. Conference on Spoken Language Processing, 2006.

[Biber1993] D. Biber, Co-occurrence Patterns Among Collocations: A Tool for Corpus-Based Lexical Knowledge Acquisition, in Computational Linguistics, 19(3), 531-538, 1993.

[Bilmes1998] J. Bilmes, A Gentle Tutorial on the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models, Technical Report ICSI-TR-97-021, University of California at Berkeley, 1997.

[Blei2004] D. M. Blei, T. L. Griffiths, M. I. Jordan and J. B. Tenenbaum, Hierarchical Topic Models and the Nested Chinese Restaurant Process, in Advances in Neural Information Processing Systems, 2004.

[Bohm2000] C. Böhm, B. Braunmüller, H-P. Kriegel and M. Schubert, Efficient Similarity Search in Digital Libraries, in Proc. of IEEE Int. Conf. Advances in Digital Libraries, 2000.

[Bohm2001] C. Böhm, S.Berchtold and D. A. Keim, Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases, in ACM Computing Survey, 33(3), 322-373, 2001.

[Bohm2004] C. Böhm and F. Krebs, The k-nearest neighbor join: Turbo charging the KDD process. in Knowledge and information systems, 6, 728-49, 2004.

[Boughorbel2005] S. Boughorbel, J. P. Tarel and N. Boujemaa, The Intermediate Matching Kernel for Image Local Features, in Proc. Intl. Joint Conf. on Neural Networks, 2005.

[Boujemaa2001] N. Boujemaa, J. Fauqueur, M. Ferecatu, F. Fleuret, V. Gouet, B. Le Saux and H. Sahbi, Ikona: interactive specific and generic image retrieval, in MMCBIR, 2001.

[Bourel2000] F. Bourel, C. C. Chibelushi and A. A. Low, Robust Facial Feature Tracking, in Proc. of British Machine Vision Conference, 1, 232-241, 2000.

[Bourgain1985] J. Bourgain and I.J. Math, On Lipschitz Embedding of Finite Metric Spaces in Hilbert Space, 52, 46-52, 1985.

[Bouteldja2006a] N. Bouteldja, V. Gouet-Brunet and M. Scholl, Back to the curse of dimensionality with local image descriptors, in CEDRIC Research Report no 1049, 2006.

[Bouteldja2006b] N. Bouteldja, V. Gouet-Brunet and M. Scholl, Evaluation of strategies for multiple sphere queries with local image descriptors, in Conference on Multimedia Content Analysis, Management and Retrieval, 1-12, 2006.

[Braunmuller2000] B. Braunmuller, M. Ester, H.-P. Kriegel and J. Sander, Efficiently supporting multiple similarity queries for mining in metric databases, in Proc. of ICDE, 256-267, 2000.

[Brill1995] E. Brill, Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging, in Computational Linguistics, 21, 543-565, 1995.

[Brin1998] S. Brin and L. Page, The anatomy of a large-scale hypertextual Web search engine, in Proc. of Conf. on World Wide Web, 107-117, 1998.

[Brown1991] P. F Brown, , S. A. Della Pietra, V.J. Della Pietra and R. L. Mercer, Word-Sense Disambiguation Using Statistical Methods, In Proc. of Meeting of the Association for Computational Linguistics, 264-270, 1991.

[Brown2001] D. Brown, I. Craw and J. Lewthwaite, A SOM Based Approach to Skin Detection with Application in Real Time Systems, in Proc. of the British Machine Vision Conference, 2001.

[Brown2005] M. Brown, R. Szeliski and S. Winder, Multi-Image Matching using Multi-Scale Oriented Patches, in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2005.

[Carmichael2003] O. Carmichael and M. Hebert, Shape-based recognition of wiry objects, in IEEE Conference On Computer Vision And Pattern Recognition, 2003.

[Carneiro2002] G. Carneiro and A. D. Jepson, Phase-based local features, in Proc. of European Conference on Computer Vision, 282–296, 2002.

[Carneiro2003] G. Carneiro and A. D. Jepson, Multi-scale phase-based local features, in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 1:736, 2003.

[Carpuat2005] M. Carpuat and D. Wu, Word Sense Disambiguation vs. Statistical Machine Translation, in Proc. of Meeting of the Association for Computational Linguistics, 387-394, 2005.

[Caspi2006] Y. Caspi, D. Simakov and M. Irani, Feature-Based Sequence-to-Sequence Matching, in International Journal of Computer Vision, 68(1), 2006.

[Cha2002] G-H. Cha, X. Zhu, P. Petkovic and C.W. Chung, An efficient indexing method for nearest neighbor searches in high-dirnensional image databases, in IEEE Transactions on Multimedia, 2002.

[Chakrabarti2000] K. Chakrabarti and S. Mehrotra, Local dimensionality reduction: A new approach to indexing high dimensional spaces, in Proc. of Int. Conf. on Very Large Data Bases, 2000.

[Chan2005] Y. S. Chan and H. T. Ng. Scaling, Up Word Sense Disambiguation via Parallel Texts, in Proc. of National Conference on Artificial Intelligence, 1037-1042, 2005.

[Chavez2001] E. Chavez, G. Navarro, R. A. Baeza-Yates and J. L. Marroquin, Searching in metric spaces, in ACM Computing Surveys, 33(3), 273-321, 2001.

[Chen1996] S.F. Chen and J. Goodman, An Empirical Study of Smoothing Techniques for Language Modeling, in Proc. of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics, Morgan Kaufmann Publishers, 310-318, 1996.

[Chen1998] S.S. Chen and P.S. Gopalakrishnan, Clustering via the Bayesian information criterion with applications in speech recognition, in Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2, 645-648, 1998.

[Chen2006a] S.F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau and G. Zweig, Advances in speech transcription at IBM under the DARPA EARS program, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1596-1608, 2006.

[Chen2006b] J. Chen, Towards High-Performance Word Sense Disambiguation by Combining Rich Linguistic Knowledge and Machine Learning Approaches, PhD Thesis, University of Pennsylvania, 2006.

[Cheung2003] S-C. Cheung and A. Zakhor, Fast Similarity Search on Video Signatures, in Proc. of ICIP, 2003.

[Church1990] K. W. Church and Patrick Hanks, Word Association Norms, Mutual Information and Lexicography, in Computational Linguistics, 16(1), 22-29, 1990.

[Ciaccia1997] P. Ciaccia, M. Patella and P. Zezula, M-tree: An efficient access method for similarity search in metric spaces, in Proc. VLDB Int. Conf., 1997.

[Ciaccia1998] P. Ciaccia, M. Patella and P. Zezula, A cost model for similarity queries in metric spaces, in Proc. of ACM SIGMOD Symp. on Principles of Database Systems, 1998.

[Ciaccia2000] P. Ciaccia and M. Patella, Pac nearest neighbor queries: Approximate and controlled search in high-dimensional and metric spaces, in Proc. of Int. Conf. on Data Engineering, 2000.

[Ciaramita2006] M. Ciaramita and Y. Altun, Broad-Coverage Sense Disambiguation and Information Extraction with a Supersense Sequence Tagger, in Proc. of EMNLP, 2006.

[Clarkson2005] K.L. Clarkson, Nearest-Neighbor Searching and Metric Space Dimensions, in Nearest-Neighbor Methods for Learning and Vision: Theory and Practice, MIT Press, Cambridge, MA 2005.

[Cognitec2007] Cognitec Systems, http://www.cognitec-systems.de/, June 2007.

[Comer1979] D. Comer, The ubiquitous B-tree, in ACM Computing Surveys, 1979.

[Cowie1992] J. Cowie, J. Guthrie and L. Guthrie, Lexical Disambiguation using Simulated Annealing, in Proc. of COLING, 359-365, 1992.

[Crandall2005] D. Crandall, P. Felzenszwalb and D. Huttenlocher, Spatial priors for part-based recognition using statistical models, in Proc. of Int. Conf. on Computer Vision and Pattern Recognition, 2005.

[Csurka2004] G. Csurka, C.R. Dance, L. Fan, J. Willamowski and C. Bray, Visual categorization with bags of keypoints, in Proc. of European Conference on Computer Vision, 2004.

[Daelemans2000], W. Daelemans, J. Zavrel, K. van der Sloot and A. van den Bosch, TiMBL: Tilburg Memory Based Learner, Technical Report 00-01 – 2000, Tilburg, 2000.

[Dagan1991] I. Dagan, A. Itai, U. Schwall, Two Languages are more informative than one, in Proc. of Annual meeting of the ACL, 130-137, 1991.

[Dai1996] Y. Dai and Y. Nakano, Face-Texture Model Based on SGLD and Its Application in Face Detection in a Color Scene, in Pattern Recognition, 29(6), 1007-1017, 1996.

[Dan2004] J. Dean and S. Ghemawat, MapReduce: Simplified Data Processing on Large Clusters, in Proc. of Symposium on Operating System Design and Implementation, 2004.

[Datar2004] M. Datar, N. Immorlica, P. Indyk and V.S. Mirrokni, Locality-sensitive hashing scheme based on p-stable distributions, in Proc. of symposium on Computational geometry, 253-262, 2004.

[Datta2006] R. Datta, W. Ge, J. Li, and J. Z. Wang, Toward bridging the annotation-retrieval gap in image search by a generative modeling approach, in Proc. ACM Multimedia, 2006.

[Datta2007] R. Datta, D. Joshi, J. Li and J. Z. Wang, `Image Retrieval: Ideas, Influences, and Trends of the New Age, in ACM Computing Surveys, 2007.

[Davis1980] S. Davis and P. Mermelstei, Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, in IEEE Transactions on Acoustics, Speech, and Signal Processing, 28(4), 357-366, 1980.

[Deibel2004] K. Deibel, Current Approaches for Unrestricted Word Sense Disambiguation, in Course notes for CSE 574, University of Washington, 2004.

[Deng2006] H. Deng, E. N. Mortensen, L. Shapiro and T. G. Dietterich, Reinforcement Matching Using Region Context, in Proc. Conf. on Computer Vision and Pattern Recognition, 2006.

[Deselaers2004] T. Deselaers, D. Keysers and H. Ney, Features for image retrieval: a quantitative comparison, in 26th DAGM Symposium Pattern Recognition, 2004.

[Diab2002] M. Diab and P. Resnik, An unsupervised method for word sense tagging using parallel corpora, in Proc. of the meeting of the ACL, 2002.

[Eickeler1999] S. Eickeler and S. Müller, Content-based video indexing of tv broadcast news using hidden Markov models, in Proc. of Int. Conf. on Acoustics, Speech, and Signal Processing, 2997-3000, 1999.

[Eisele1996] T. Eisele, H.R. Umbach and D. Langmann, A Comparative Study of Linear Feature Transformation Techniques for Automatic Speech Recognition, in Proc. of ICSLP, 252-255, 1996.

[Everingham2006] M. Everingham, A. Zisserman, C.K.I. Williams and L. VanGool, The PASCAL Visual Object Classes Challenge Results VOC 2006, Tech. Report, http://www.pascalnetwork.org/challenges/VOC/voc2006/results.pdf

[FaceIt2007] FaceIt Argus, http://www.l1id.com/, June 2007.

[FaceSnap2007] FaceSnap Recorder, http://www.facesnap.de/htd/fsnapr.html, June 2007.

[Fagin2003] R. Fagin, R. Kumar and D. Sivakumar, Efficient similarity search and classification via rank aggregation, in Proc. of SIGMOD, 301-312, 2003.

[Faloutsos1995] C. Faloutsos and K.-I. Lin, FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Traditional and Multimedia Datasets, in Proc. of ACM SIGMOD Int. Conf. on Management of Data, 1995.

[Faloutsos1996] C. Faloutsos, Searching Multimedia Databases by Content, in Kluwer Int. series on advances in database systems, 3, 1996.

[Fauqueur2004] J. Fauqueur and N. Boujemaa, Region-based image retrieval: Fast coarse segmentation and fine color description, in Journal of Visual Languages and Computing, 15(1), 69-95, 2004.

[Fellbaum1998] C. Fellbaum, WordNet, an Electronic Lexical Database, Cambridge/London: MIT Press, 1998.

[Feltzenswalb2005] P. Feltzenswalb and D. Hutenlocher, Pictorial structures for object recognition, in Int. Journal on Computer Vision, 61, 55-79, 2005.

[Ferecatu2005] M. Ferecatu, Image retrieval with active relevance feedback using both visual and keyword-based descriptors, PhD thesis, University of Versailles Saint-Quentin-En-Yvelines, 2005.

[Fergus2005] R. Fergus, P. Perona, and A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in Int. Journal on Computer Vision, 2005.

[Fergus2005b] R. Fergus, P. Perona, and A. Zisserman, A sparse object category model for efficient learning and exhaustive recognition, in Proc. of CVPR, 2005.

[Ferhatosmanoglu2000] H. Ferhatosmanoglu, E. Tuncel, D. Agrawal and A. E. Abbad, Vector Approximation based Indexing for Non-uniform High Dimensional Data Sets, in Proc. of Int. Conf. on Information and Knowledge Management, 2000.

[Ferhatosmanoglu2001] H. Ferhatosmanoglu, E. Tuncel, D. Agrawal and A. El Abbadi, Approximate nearest neighbor searching in multimedia databases, in Proc. of Int. Conf. on Data Engineering, 503–514, 2001.

[Finlayson1996] G. Finlayson, Color in perspective, in IEEE Trans. Pattern Analysis and Machine Intelligence, 18(10), 1034–1038, 1996.

[Fischler1973] M.A. Fischler and R.A. Elschlager, The representation and matching of pictorial structures, in IEEE Transactions on Computer, 22(1), 1973.

[Flickner1995] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, Query by image and video content: The qbic system, in IEEE Computer, 28(9), 23–32, 1995.

[Florack1994] L. M. J. Florack, B. M. ter Haar Romeny, J. J. Koenderink, and M. A. Viergever, General intensity transformations and differential invariants, in JMIV, 4(2), 171-187, 1994.

[Freeman1991] W.T. Freeman and E.H. Adelson, The design and use of steerable filters, 13(9), 891-906, 1991.

[Gaede1998], V. Gaede and O. Günther, Multidimensional access methods, in ACM computing surveys, 30(2), 170-231, 1998

[Gale1992] W. Gale, K. W. Church and D. Yarowsky, A Method for Disambiguating Word Senses in a Large Corpus, in Computers and the Humanities, 26(5-6), 415-439, 1992.

[Gales2006] M.J.F. Gales, Y. Do, P.C. Woodland, D. Ho Yin Chan Mrva, R. Sinha and S.E. Tranter, Progress in the CU-HTK broadcast news transcription system, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1513-1525, 2006.

[Gao2004] D. Gao and N. Vasconcelos, Discriminant saliency for visual recognition from cluttered scenes, in Journal on Advances in Neural Information Processing Systems, 2004.

[Garside1987] R. Garside, G. Leech and G. Sampson, The Computational Analysis of English, a Corpus-Based Approach, London: Longman, 1987.

[Gerbrands1981] J. Gerbrands, On the relationships between SVD, KLT and PCA, in Pattern Recognition, 14, 375-381, 1981.

[Gevers2000] T. Gevers and A. Smeulders, Pictoseek: Combining color and shape invariant features for image retrieval, in IEEE Trans. Image Processing, 9(1), 102-119, 2000.

[Gilles1998] S. Gilles, Robust Description and Matching of Images, PhD thesis, Oxford University, 1998.

[Gionis1999] A. Gionis, P. Indyk and R. Motwani, Similarity Search in High Dimensions via Hashing, in VLDB Journal, 1999.

[Goldstein2000] J. Goldstein and R. Ramakrishnan, Contrast plots and p-sphere trees: Space vs. time in nearest neighbor searches, in Proc. of Int. Conf. on Very Large Data Bases, 2000.

[Gool1996] L. J. Van Gool, T. Moons and D. Ungureanu, Affine photometric invariants for planar intensity patterns, in Proc. of European Conference on Computer Vision, 1, 642–651, 1996.

[Grauman2005a] K. Grauman and T. Darrell, Efficient Image Matching with Distributions of Local Invariant Features, In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR '05), 2005.

[Grauman2005b] K. Grauman and T. Darrell, The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features, in Proc. IEEE Conf. on Computer Vision, 2005.

[Grigorescu2002] S. Grigorescu, N. Petkov, and P. Kruizinga, Comparison of texture features based on gabor filters, in IEEE Trans. on Image Processing, 11(10), 1160–1167, 2002.

[Grira2006] N. Grira, M. Crucianu and N. Boujemaa, Fuzzy Clustering With Pairwise Constraints for Knowledge-Driven Image Categorization, in Vision, Image & Signal Processing, 2006.

[Gu2003] H. Gu, G. Su and C. Du, Feature points extraction from faces, in D. G. Bailey, editor, Image and Vision Computing, pages 154-158, 2003.

[Gunn1998] S. Gunn and M. Nixon, Global and Local Active Contours for Head Boundary extraction, in Int. Journal on Computer Vision, 30, 1998.

[Guttman1984] A. Guttman, R-Trees: A Dynamic Index Structure for Spatial Searching, in Proc. of SIGMOD Conf., 47-57, 1984.

[Hadoop] Hadoop, http://lucene.apache.org/hadoop/

[Harris1988] C. Harris and M. Stephens, A combined corner and edge detection, In Proc. of Alvey Vision Conference, 147–151, 1988.

[Heisele2001] B. Heisele, T. Serre, M. Pontil and T. Poggio, Component-based Face Detection, in Proc. of CVPR, 2001.

[Henrich1989] A. Henrich, H-W. Six and P. Widmayer, The LSD tree: Spatial Access to Multidimensional Point and Nonpoint Objects, in Proc. of VLDB Conf.,: 45-53, 1989.

[Henrich1996] A. Henrich, The LSD-h-Tree: An Access Structure for Feature Vectors, in Proc. of Int. Conf. on Data Engineering, 1998.

[Hermansky1991] H. Hermansky and L.A. Cox, Perceptual Linear Predictive (PLP) Analysis-Resynthesis Technique, in Proc. of IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics, 37-38, 1991.

[Herve2007] N. Hervé and N. Boujemaa, Image annotation: which approach for realistic databases ?, ACM Int. Conf. on Image and Video retrieval, 2007.

[Hess2007] R. Hess and A. Fern, Improved Video Registration using Non-Distinctive Local Image Features, in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2007.

[Hillel2005] A. B. Hillel, T. Hertz, and D.Weinshall, Efficient learning of relational object class models, in IEEE Conf. on Computer Vision, 2005.

[Hjaltason1995] G. R. Hjaltason and H. Samet, Ranking in spatial databases, in Proc. of Advances in Spatial Databases Symposium, 83-95, 1995.

[Hjaltason2003] G. R. Hjaltason and H. Samet, Properties of embedding methods for similarity searching in metric spaces, in IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5), 530-549, 2003.

[Hjelmas2001] E. Hjelmas and B. Low, Face Detection: A Survey, in Computer Vision and Image Understanding, 83(3), 236-274, 2001.

[Holub2005] A. Holub and P. Perona, A discriminative framework for modeling object classes, in Int. Conf. on Computer Vision, 2005.

[Houle2005] M. Houle and J. Sakuma, Fast approximate similarity search in extremely high-dimensional data sets, in Int. Conf. on Data Engineering, 619-630, 2005.

[Hua2002] R. Hua, L. De Silva and V. Prahlad, Detection and tracking of faces in real-time environments, in Proc. of Int. Conf. on Imaging Science, Systems and Technology, 2002.

[Huang1999] J. Huang, S. Ravi Kumar, M. Mitra, W.-J. Zhu and R. Zabih, Spatial color indexing and applications, in Int. Journal on Computer Vision, 35(3), 245-268, 1999.

[Huang2001] X. Huang, A. Acero, and H. Hon, Spoken Language Processing: A Guide to Theory, in Algorithm and System Development, Prentice Hall PTR, 2001.

[Hug2000] M. Hug, Partial Disambiguation for Very Ambiguous Grammatical Words, in Journal of Quantitative Linguistics, 7(3), 217-226, 2000.

[Indyk2000] P. Indyk, Dimensionality Reduction Techniques for Proximity Problems, in Symposium on Discrete Algorithms, 2000.

[James1995] A. James, Natural Language Understanding, Redwood City, CA: Benjamin Cummings, 1995.

[Joachims1999] T. Joachims, Making Large-Scale SVM Learning practical, in Advances in Kernel Methods - Support vector learning, 169-184, Cambridge MA: MIT Press, 1999.

[Johnson1997] A. E. Johnson and M. Hebert, Recognizing objects by matching oriented points, in Proc. of the Conference on Computer Vision and Pattern Recognition, 1997.

[Johnson1999] A. Johnson and M. Hebert, Using Spin Images for Efficient Object Recognition in Cluttered 3D Scenes, in IEEE Trans. PAMI 21(5), 433-449, 1999.

[Johnson2000] S.E. Johnson and P.C. Woodland, A method for direct audio search with applications to indexing and retrieval, in Proc. of ICASSP, 3, 1427-1430, vol.3, 2000.

[Joly2004] A. Joly, C. Frélicot and O. Buisson, Feature statistical retrieval applied to content-based copy identification, in Proc. of Int. Conf. on Image Processing, 2004.

[Joly2007] A. Joly, O. Buisson and C. Frélicot, Content-Based Copy Retrieval Using Distorsion-based Probabilistic Similarity Search, in IEEE Transactions on Multimedia, 2007.

[Joshi2006a] M. Joshi, S. Pakhomov, T. Pedersen, R. Maclin and C. Chute, An End-to-End Supervised Target-Word Sense Disambiguation System, in Proc. of National Conference on Artificial Intelligence, 2006.

[Joshib2006b] M. Joshi, T. Pedersen, R. Maclin and S. Pakhomov, Kernel Methods for Word Sense Disambiguation and Acronym Expansion, in Proc. of National Conference on Artificial Intelligence, 2006.

[Jurie2004] F. Jurie and C. Schmid, Scale-invariant shape features for recognition of object categories, in Int. Conf. CVPR, 2, 90-96, 2004.

[Kadir2001] T. Kadir and M. Brady, Saliency, scale and image description, in int. Journal on Computer Vision, 45(2), 83-105, 2001.

[Katayama1997] N. Katayama and S'I. Satoh, The SR-tree: An Index Structure for High-Dimensional Nearest Neighbor Queries, in Proc. of ACM SIGMOD Conf., 1997.

[Ke2004a] Y. Ke, R. Sukthankar and L. Huston, Efficient Near-Duplicate Detection and Sub-Image Retrieval, In Proc. ACM Conf. on Multimedia, 2004.

[Ke2004b] Y. Ke and R. Sukthankar, Pca-sift: A more distinctive representation for local image descriptors, in Proc. of CVPR, 2004.

[Kilgarriff2000] A. Kilgarriff and M. Palmer, Guest Editors of the Special Issue on SENSEVAL, Computers and the Humanities, 34, 127-134, 2000.

[Kim2006] S. Kim, K. Yoon and I. Kweon, Background Robust Object Labeling by Voting of Weight-Aggregated Local Features, in Proc. of. Int. Conf. on Pattern Recognition, 2006.

[Klabbers2001] E. Klabbers, K. Stober, R. Veldhuis, P. Wagner and S. Breuer, Speech synthesis development made easy: The Bonn Open Synthesis System, in Proc. of Eurospeech Conference, 521-524, 2001.

[Kleinberg1997] J. M. Kleinberg, Two algorithms for nearest-neighbor search in high dimensions, in Proc. of ACM symposium on Theory of computing, 599-608, 1997.

[Kleinberg1999] J.M. Kleinberg, Authoritative Sources in a Hyperlinked Environment, in Journal of the ACM, 46(5): 604-632, 1999.

[Koenderink1984] J. Koenderink, The structure of images, in Biological Cybernetics, 50(5), 363-370, 1984.

[Koenderink1987] J. Koenderink and A. J. van Doom, Representation of local geometry in the visual system, in Biological Cybernetics, 55(6), 367–375, 1987.

[Kohavi1997] R. Kohavi and G. H. John, Wrappers for feature subset selection, in Artificial Intelligence, 1, 1997.

[Kohomban2005] U. S. Kohomban and W. S. Lee, Learning Semantic Classes for Word Sense Disambiguation, in Proc. of Annual Meeting of the ACL, 34-41, 2005.

[Korn2001] F. Korn, B. Pagel and C. Faloutsos, On the dimensionality curse and the self-similarity blessing, in IEEE Trans. on Knowledge and Data Engineering, 2001.

[Kotropoulos1997] C. Kotropoulos and I. Pitas, Rule-Based Face Detection in Frontal Views, in Proc. of Int. Conf. on Acoustics, Speech and Signal Processing, 4, 2537-2540, 1997.

[Kuropka2004] D. Kuropka, Modelle zur Repräsentation natürlichsprachlicher Dokumente Information Filtering und Retrieval mit relationalen Datenbanken, in Advances in Information Systems and Management Science, 2004.

[Lamel1996] L. Lamel and G. Adda, On Designing Pronunciation Lexicons for Large Vocabulary, Continuous Speech Recognition, in Proc. of ICSLP, 6-9, 1996.

[Landes1998] S. Landes, C. Leacock and R Tengi, Building Semantic Concordances, in WordNet: An Electronic Lexical Database, MIT Press, Cambridge, MA, 1998.

[Laptev2003] I. Laptev and T. Lindeberg, Space-time Interest Points, in IEEE Int. Conf. on Computer Vision, 2003.

[Laptev2004] I. Laptev and T. Lindeberg, Local Descriptors for Spatio-Temporal Recognition, in Proc. of ECCV, 2004.

[Laptev2005] I. Laptev, On Space-Time Interest Points, in Int. Journal of Computer Vision, 64, 2005.

[Larson2003] M. Larson and S. Eickeler, Using syllable-based indexing features and language models to improve german spoken document retrieval, in Proc. of European Conference on Speech Communication and Technology, 2003.

[Law2006] J. Law-To, O. Buisson, V. Gouet-Brunet and N. Boujemaa, Robust Voting Algorithm Based on Labels of Behavior for Video Copy Detection, in Proc. of the ACM Multimedia, 2006.

[Law2007] J. Law-To, L. Chen, A. Joly, I. Laptev, O. Buisson, V. Gouet-Brunet, N. Boujemaa and F. Stentiford, Video Copy Detection: a Comparative Study, in Proc. of the ACM International Conference on Image and Video Retrieval, 2007.

[Lawder2000a] J.K. Lawder and P.J.H. King, Using Space-filling Curves for Multi-dimensional Indexing, in Proc. of British National Conf. on Databases: Advances in Databases, 2000.

[Lawder2000b] J.K. Lawder, Calculation of Mappings Between One and n-dimensional Values Using the Hilbert Space-filling Curve, in Technical Report JL1/00, Birkbeck College, University of London, 2000.

[Lawder2001] K. Lawder and P.J.H. King, Querying Multi-dimensional Data Indexed Using the Hilbert Space-Filling Curve, in SIGMOD Record, 2001.

[Lazebnik2003] S. Lazebnik, C. Schmid, and J. Ponce, Sparse texture representation using affine-invariant neighborhoods, in Proc. of CVPR, 2003.

[Lazebnik2004] S. Lazebnik, C. Schmid and J. Ponce, Semi-Local Affine Parts for Object Recognition, in BMVC, 2004.

[Lazebnik2006] S. Lazebnik, C. Schmid and J. Ponce, Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2006.

[Leacock1998] M. Leacock, M. Chodorow, and G. A. Miller, Using corpus statistics and WordNet relations for sense identification, in Computational Linguistics, 24, 147–165, 1998.

[Leibe2004] B. Leibe, A. Leonardis and B. Schiele, Combined object categorization and segmentation with an implicit shape model, in Proc. of ECCV Workshop on statistical learning in computer vision, 2004.

[Lejsek2005] H. Lejsek, F-H. Ásmundsson, B. Pór-Jónsson and L. Amsaleg, Efficient and Effective Image Copyright Enforcement, in Journées Bases de Données Avancées, 2005.

[Lejsek2006] H. Lejsek, F-H. Ásmundsson, B. Pór-Jónsson and L. Amsaleg, Scalability of Local Image Descriptors: A Comparative Study, in Proc. of ACM Int. Conf. on Multimedia, 2006.

[Lepetit2005] V. Lepetit, P. Lagger and P. Fua, Randomized Trees for Real-Time Keypoint Recognition, in Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, 2005.

[Lesk1986] M. Lesk, Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a Pine Cone from an Ice Cream Cone, in Proc. of the SIGDOC Conference, 24-26, 1986.

[Leung2001] T. Leung and J. Malik, Representing and recognizing the visual appearance of materials using three-dimensional textons, in Int. Journal on Computer Vision, 43(1), 29-44, 2001.

[Li1999] C. Li, E. Chang, H. Garcia-Molina, J. Wang and G. Wiederhold, Clindex: Clustering for Similarity Queries in High-Dimensional Spaces, Technical Report SIDL-WP-1998-0100, Stanford University, 1999.

[Li2002] C. Li, E. Chang, H. Garcia-Molina and G. Wiederhold, Clustering for Approximate Similarity Search in High-Dimensional Spaces, in IEEE Transactions on Knowledge and Data Engineering, 2002.

[Li2004] H. Li and C. Li 2004, Word Translation Disambiguation using bilingual bootstrapping, in Computational Linguistics 20(4), 563-596, 2004.

[Li2006] J. Li and J. Z. Wang, Real-time Computerized Annotation of Pictures, in Proc. of the ACM Multimedia Conference, 911-920, 2006.

[Lin1994] K. I. Lin, H. V. Jagadish and C. Faloutsos, The tv-tree: an index structure for highdimensional data, in VLDB Journal, 3(4), 517-542, 1994.

[Lin1998] D. Lin, Automatic retrieval and clustering of similar words, in Proc. of COLING-ACL conference, 1998.

[Lindeberg1994] T. Lindeberg, Scale-space theory in computer vision, Monograph, Kluwer, Dordrecht, 1994.

[Lindeberg1998] T. Lindeberg, Feature detection with automatic scale selection, in Int. Journal of Computer Vision, 30(2), 77-116, 1998.

[Liu2004] T. Liu, A. Moore, A. Gray and K. Yang, An Investigation of Practical Approximate Nearest-Neighbor Algorithms, in Proc. of NIPS, 2004.

[Liu2007] T. Liu, C. Rosenberg and H. A. Rowley, Clustering Billions of Images with Large Scale Nearest Neighbor Search, in Proc. of IEEE Work. on Applications of Computer Vision, 2007.

[Loupias2000] E. Loupias and N. Sebe, Wavelet-based salient points: Applications to image retrieval using color and texture features, In Visual Information and Information Systems, 223–232, 2000.

[Lowe1999] D. G. Lowe, Object recognition from local scale-invariant features, In Proc. of Int. Conference on Computer Vision, 1150-1157, 1999.

[Lowe2001] D. Lowe, Local feature view clustering for 3d object recognition, in Proc. of CVPR, 682-688, 2001.

[Lowe2004] D. G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, in Int. Journal of Computer Vision, 60(2), 2004.

[Loy2003] G. Loy and A. Zelinsky, Fast radial symmetry for detecting points of interest, in IEEE Trans. on Pattern Analysis and Machine Intelligence, 25(8), 959–973, 2003.

[Lu2004] X. Lu and R. Manduchi, Wide Baseline Feature Matching Using the Cross-Epipolar Ordering Constraint, In Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 2004.

[Lv2006] Q. Lv, W. Josephson, Z. Wang, M. Charikar and K. Li, A Time-Space Efficient Locality Sensitive Hashing Method for Similarity Search in High Dimensions, in Princeton Technical Report TR-759-06, 2006.

[Malciu2000] M. Malciu and F. Preteux, Tracking facial features in video sequences using a deformable model-based approach, in Proc. of SPIE Conf. on Mathematical Modeling, Estimation and Imaging, 51-62, 2000.

[Mallat1989] S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, in IEEE Trans. Pat. Anal. Mach. Intell., 11, 674–693, 1989.

[Manjunath2001] B. S. Manjunath, J-R. Ohm, V. Vasudevan, A. and Yamada, Color and texture descriptors, in IEEE Trans. on Circuits and Systems for Video Technology, 11(6), 703-715, 2001.

[Manning1999] C. Manning and H. Schütze, Word Sense Disambiguation, in Foundations of Statistical Natural Language Processing, Cambridge, Massachusetts, MIT Press, 229-314, 1999.

[Manning2007] C.D. Manning, P. Raghavan and H. Schütze, Information Retrieval, Cambridge UP, 2007.

[Manolopoulos2003] Y. Manolopoulos, A. Nanopoulos, A. N. Papadopoulos, and Y. Theodoridis, R-trees have grown everywhere, in ACM Computing Surveys, 2003.

[Martinez2002] J.M. Martínez, R. Koenen and F. Pereira, MPEG-7: the generic Multimedia Content Description Standard, in IEEE Multimedia, 9(2), 78-87, 2002.

[Martinez2006] D. Martinez, E. Agirre, X-L. Wang, Word Relatives in Context for WSD, in Proc. of the Australasian Language Technology Workshop, 42-50, 2006.

[Matas2002] J. Matas, O. Chum, U. Martin, and T. Pajdla, Robust wide baseline stereo from maximally stable extremal regions, in Proc. of the British Machine Vision Conference, 1, 384-393, 2002.

[Matsoukas2006] S. Matsoukas, J.L. Gauvain, G. Adda, T. Colthurst, C. Kao, O. Kimball, L. Lamel, F. Lefevre, J.Z. Ma, J. Makhoul, L. Nguyen, R. Prasad, R. Schwartz, H. Schwenk and B. Xiang, Advances in transcription of broadcast news and conversational telephone speech within the combined EARS BBN/LIMSI system, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1541-1556, 2006.

[McCarthy2004] D. McCarthy, R. Koeling, J. Weeds and J. Carroll 2004, Finding Predominant Word Senses in Untagged Text, in Proc. of Annual Meeting of the ACL, 2004.

[McInnes2007] B. T. McInnes, Accurate and Scalable Word Sense Disambiguation in the Biomedical Domain, PhD Thesis, 2007.

[Mctait2003] K. Mctait and M. Adda-Decker, The 300k LIMSI German Broadcast News Transcription System, in Proc. of European Conference on Speech Communication and Technology, 2003.

[Mihalcea1999] R. Mihalcea and D. Moldovan, A method for word sense disambiguation of unrestricted text, in Proc. of the meeting of the ACL, 1999.

[Mihalcea2003] R. Mihalcea, The role of non-ambiguous words in natural language disambiguation, in Proc.of Conf. on recent advances in natural language processing, 2003.

[Mihalcea2004] R. Mihalcea, T. Chklovski and A. Kilgariff, The senseval-3 lexical sample task, in Proc. of ACL workshop on the Evaluation of Systems for the Semantic Analysis of Text, 2004.

[Mikolajczyk2001] K. Mikolajczyk and C. Schmid, Indexing based on scale invariant interest points, in Proc. of ICCV, 525-531, 2001.

[Mikolajczyk2002] K. Mikolajczyk and C. Schmid, An affine invariant interest point detector, in Proc. of ECCV, 128-142, 2002.

[Mikolajczyk2003] K. Mikolajczyk, A. Zisserman and C. Schmid, Shape recognition with edge-based features, in British Machine Vision Conference, 2, 779-788, 2003.

[Mikolajczyk2004] K. Mikolajczyk and C. Schmid, Scale & Affine Invariant Interest Point Detectors, in International Journal of Computer Vision, 60(1), 2004.

[Mikolajczyk2005a] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L. Van Gool, A comparison of affine region detectors, in Int. Journal on Computer Vision, 65(1-2), 43–72, 2005.

[Mikolajczyk2005b] K. Mikolajczyk and Cordelia Schmid, A performance evaluation of local descriptors, in IEEE Transactions on Pattern Analysis & Machine Intelligence, 27(10), 1615–1630, 2005.

[Miller1990] G. Miller, WordNet: an On-line lexical database, in Int. Journal of Lexicography, 3(4), 235-312, 1990.

[Miller2002] M. L. Miller, M. A. Rodriguez and I. J. Cox, Audio fingerprinting: nearest neighbor search in high dimensional binary spaces, in IEEE Workshop on Multimedia Signal Processing, 2002.

[Mindru1992] F. Mindru, T. Moons, and L. Van Gool, Recognizing color patterns irrespective of viewpoint and illumination, in Conference on Computer Vision and Pattern Recognition, 368-373, 1992.

[Mindru2004] F. Mindru, T. Tuytelaars, L. Van Gool and Theo Moons, Moment invariants for recognition under changing viewpoint and illumination, in Computer Vision and Image Understanding, 94(1-3), 3-27, 2004.

[Moellic2006] P.A. Moellic and C. Fluhr., Imageval 2006 official campaign, Technical report, CEA List, 2006.

[Moenne2006] N. Moenne-Loccoz, E. Bruno and S. Marchand-Maillet, Feature Trajectories for Efficient Event-Based Indexing of Video Sequences, in Proc. of Int. Conf. on Image and Video Retrieval, 2006.

[Mokhtarian1996] F. Mokhtarian, S. Abbasi, and J. Kittler, Robust and efficient shape indexing through curvature scale space, in Vision Speech and Signal Processing, 1996.

[Montesinos1998] P. Montesinos, V. Gouet and R. Deriche, Differential invariants for color images, in Proc. Int. Conference on Pattern Recognition, 1998.

[Moore2000] A. W. Moore, The Anchors Hierarchy: Using the Triangle Inequality to Survive High Dimensional Data, in Proc. of Conf. on Uncertainty in Artificial intelligence, 397-405, 2000.

[Murphy2006] K. Murphy, A. Torralba, D. Eaton and W. Freeman, Object Detection and Localization Using Local and Global Features, in Proc. of Conf. Toward Category-Level Object Recognition, 382-400, 2006.

[Navigli2007] R. Navigli and M. Lapata, Graph Connectivity Measures for Unsupervised Word Sense Disambiguation, in Proc. of the Int. Joint Conference on Artificial Intelligence, 1683-1688, 2007.

[Ney1997] H. Ney and S. Ortmanns, Progress in Dynamic Programming Search for LVCSR, in Proc. of the IEEE, 1224-1240, 1997.

[Ng1998] K. Ng and V. Zue, Phonetic Recognition for Spoken Document Retrieval, in Proc. of ICASSP, 325-328, 1998.

[Ng2003] H. T. Ng, B. Wang, Y. S. Chan, Exploiting Parallel Texts for Word Sense Disambiguation: An Empirical Study, in Proc. of ACL, 455-462, 2003.

[Nister2005] D. Nistér, Preemptive RANSAC for Live Structure and Motion Estimation, in Machine Vision and Applications, 16(5), 2005.

[Novak2006] D. Novak and P. Zezula, M-Chord: a scalable distributed similarity search structure, in Proc. of the Int. Conf. on Scalable information Systems, 2006.

[Ohtsuki2006] K. Ohtsuki, K. Bessho, Y. Matsuo, S. Matsunaga and Y. Hayashi, Automatic multimedia indexing: combining audio, speech, and visual information to index broadcast news, in IEEE Signal Processing Magazine, 23(2), 69-78, 2006.

[Ojala2002] T. Ojala, M. Pietikäinen, and T. Maenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, in IEEE Trans. Pattern Anal. Mach. Intell., 24(7), 971-987, 2002.

[Opelt2006] A. Opelt, M. Fussenegger, A. Pinz and P. Auer, Generic Object recognition with boosting, in PAMI, 2006.

[Osuna1997] E. Osuna, R. Freund and F. Girosi, Training support vector machines: an application to face detection, in Proc. of IEEE Conf. on Computer Vision and Pattern Recognition, 1997.

[Page1999] L. Page, S. Brin, R. Motwani and T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, in Technicl Report SIDL-WP-1999-0120, Stanford Digital Library Technologies Project, Stanford CA, 1999.

[Pagel2000] B-U. Pagel, F. Korn and C. Faloustos, Deflating the dimensionality curse using multiple fractal dimensions, in Proc. of Int. Conf. on Data Engineering, 2000.

[Panigrahy2006] R. Panigrahy, Entropy based nearest neighbor search in high dimensions, in Proc. of ACM-SIAM Symposium on Discrete Algorithms, 2006.

[Patwardhan2003] S. Patwardhan and T. Pedersen, The cpan wordnet similarity package, http://search.cpan.org/author/SID/WordNet-Similarity-0.03/.

[Perlibakas2003] V. Perlibakas, Automatical detection of face features and exact face contour, in Pattern Recognition Letters, 24(16), 2977-2985, 2003.

[Pinquier2004] J. Pinquier and R. Andre-Obrecht, Jingle detection and identification in audio documents, in Proc. of ICASSP, 329-332, 2004.

[Platel2005] B. Platel, E. Balmachnova, L. M. J. Florack, F. M. W. Kanters and B. M. ter Haar Romeny, Using Top-Points as Interest Points for Image Matching, In Proc. 1st Intl. Workshop on Deep Structure, Singularities and Computer Vision, 2005.

[Poullot2007] S. Poullot, O. Buisson and M. Crucianu, Zgrid-based Probabilistic Retrieval for Scaling Up Content-Based Copy Detection, in Proc. of CIVR, 2007.

[Rebai2006] A. Rebai, A. Joly, and N. Boujemaa, Constant tangential angle elected interest points, in Proc. of ACM Int. Work. on Multimedia Information Retrieval, 203–212, 2006.

[Robertson1998] S. E. Robertson, S. Walker and M. Hancock-Beaulieu, Okapi at TREC-7, in Proc. of Text Retrieval Conference, 1998.

[Robinson1981] J. T. Robinson, The K-D-B-Tree: A Search Structure For Large Multidimensional Dynamic Indexes, in Proc. of ACM SIGMOD Conf., 1981.

[Roussopoulos1995] N. Roussopoulos, S. Kelley and F. Vincent, Nearest neighbor queries, in Proc. of ACM Sigmod, 1995.

[Rowley1998] H. Rowley, S. Baluja and T. Kanade, Neural Network-Based Face Detection, in IEEE Transactions on Pattern Analysis and Machine Intelligence, 20, 22-38, 1998.

[Saita2006] C-A. Saita, Cost-based query-adaptive clustering for multidimensional objects with spatial extents, PhD Thesis, Université de Versailles Saint-Quentin-en-Yvelines, 2006.

[Salembier2002] P. Salembier and T. Sikora, Introduction to MPEG-7: Multimedia Content Description Interface, John Wiley & Sons, Inc., 2002.

[Salton1975] G. Salton, A. Wong and C. S. Yang, A Vector Space Model for Automatic Indexing, Communications of the ACM, 18(11), 613-620, 1975.

[Samet2006] H. Samet, Foundations of Multidimensional and Metric Data Structures, Morgan-Kaufmann, San Francisco, 2006.

[Sanderson2000] M. Sanderson, Retrieving with Good Sense, in Information Retrieval, 2(1), 49-69, 2000.

[Sankaranarayanan2007] J. Sankaranarayanan, H. Samet and A. Varshney, A fast all nearest neighbor algorithm for applications involving large point-clouds, in Computer and Graphics, 31(2), 157-174, 2007.

[Saon2000] G. Saon, M. Padmanabhan, R. Gopinath and S. Chen, Maximum likelihood discriminant feature spaces, in Proc. of ICASSP, 1129-1132, 2000.

[Schaffalitzky2002] F. Schaffalitzky and A. Zisserman, Multi-view matching for unordered image sets, or “How do I organize my holiday snaps?, in Proc. of ECCV, vol. 1, 414-431, 2002.

[Schaffalitzky2005] F. Schaffalitzky and A. Zisserman, Multi-view matching for unordered image sets, or “How do I organize my holiday snaps?, in Proc. of ECCV, 1, 414-431, 2002.

[Schmid2001] C. Schmid, Constructing models for content-based image retrieval, in IEEE Conf. on Computer Vision and Pattern Recognition, 2, 2001.

[Schneider07] D. Schneider, T. Winkler, J. Löffler and J. Schon, Proceedings of Mobile Response 2007: International Workshop on Mobile Information Technology for Emergency Response, Springer, Berlin, 2007.

[Sebe2000] N. Sebe, M. S. Lew, Q. Tian, T. S. Huang, and E. Loupias, Color indexing using wavelet-based salient points, in Proc. of the IEEE Work. on Content-based Access of Image and Video Libraries, 2000.

[Sebe2002] N. Sebe, Q. Tian, E. Loupias, M. S. Lew and T. S. Huang, Evaluation of salient point techniques, in Proc. of the International Conference on Image and Video Retrieval, 367–377, 2002.

[Sellis1987] T. Sellis, N. Roussopoulos and C. Faloutsos The R+-Tree: A Dynamic Index For Multi-Dimensional Objects, in VLDB Journal, 1987.

[Shann1984] P. Shann, Machine Translation: A problem of Linguistic Engineering or of Cognitive Modelling?, in Proc. of the 3rd Lugano Tutorial, Edinburgh University Press, 71-90, 1984.

[Shao2007] J. Shao, Z. Huang, H. T. Shen, X. Zhou and Y. Li, Dynamic Batch Nearest Neighbour Search in Video Retrieval, in Proc. of IEEE Int. Conf. on Data Engineering, 2007.

[Shi1994] J. Shi and C. Tomasi, Good features to track, in Proc. of CVPR, 1994.

[Shi2006] X. Shi, A.L. Ribeiro Castro, R. Manduchi, and R. Montgomery, Rotational invariant operators based on steerable filter banks, in IEEE Signal Processing Letters, 13(11), 684–687, 2006.

[Singhal1996] A. Singhal, G. Salton, M. Mitra and C. Buckley, Document length normalization, in Information Processing and Management, 32, 619-633, 1996.

[Sivic2005] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman and W. T. Freeman, Discovering object categories in image collections, in Proc. of ICCV, 2005.

[Sivic2006] J. Sivic, F. Schaffalitzky and A. Zisserman, Efficient object retrieval from videos, PhD Thesis, University of Oxford, 2006.

[Smeulders2000] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, Content-based image retrieval at the end of the early years, in IEEE Trans. Pattern Analysis and Machine Intelligence 22(12), 1349-1380, 2000.

[Smith1997] J. Smith and S-F. Chang, Visualseek: a fully automated content-based image query system, in Proc. of ACM Multimedia, 1997.

[Snyder2004] B. Snyder and M. Palmer, The English All-Words Task, in Proc. of Int. Workshop on the Evaluation of Systems for the Semantic Analysis of Text, 41-43, 2004.

[Sparck1998] K. Sparck Jones, S. Walker and S. E. Robertson, A probabilistic model of information retrieval: Development and status, Technical Report 446, Cambridge University Computer Laboratory, 1998.

[Stevenson2001] M. Stevenson and Y. Wilks, Interaction of Knowledge Sources in Word Sense Disambiguation, in Computational Linguistics, 27(3), 321-349, 2001.

[Stokoe2003] C. Stokoe, M. P. Oakes and J. Tait, Word Sense Disambiguation in Information Retrieval Revisited, in Proc. of SIGIR, 159- 166, 2003.

[Stokoe2005] C. Stokoe, Differentiating Homonymy and Polysemy in Information Retrieval, in Proc. of Human Language Technology Conference, 403-410, 2005.

[Stolcke2006] A. Stolcke, B. Chen, H. Franco, H. Venkata, M. Graciarena, M. Hwang, K. Kirchhoff, A. Mandal, N. Morgan, X. Lei, T. Ng, M. Ostendorf, K. Sonmez, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng and Q. Zhu, Recent innovations in speech-to-text transcription at SRI-ICSI-UW, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1729-1744, 2006.

[Stottinger2007] J. Stöttinger, N. Sebe, T. Gevers and Allan Hanbury, Colour interest points for image retrieval, in Computer Vision Winter Workshop, 2007.

[Sudderth2005] E. B. Sudderth, A. Torralba, W. T. Freeman and A. S. Willsky, Learning Hierarchical Models of Scenes, Objects, and Parts, in Proc. of IEEE Int. Conf. on Computer Vision, 2, 2005.

[Sung1998] K. Sung and T. POGGIO, Example-Based Learning for View-Based Human Face Detection, in IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(1), 1998.

[Sussna1993] M. Sussna, Word Sense Disambiguation for Free-Text Indexing Using a Massive Semantic Network in Proc. of Int. Conf. on information and knowledge management, 1993.

[Swain1991] M. Swain and B. Ballard, Color indexing, in Int. Journal on Computer Vision, 7(1), 11-32, 1991.

[Szumilas2007] L. Szumilas, R. Donner, G. Langs and A. Hanbury, Local structure detection with orientation-invariant radial configuration, in Proc. of CVPR 2007.

[Teh2006] Y.W. Teh, M.I. Jordan, M.J. Beal and D.M. Blei, Hierarchical Dirichlet processes, in Journal of the American Statistical Association, 2006.

[Thomas1996] J. Thomas and A. Wilson, Methodologies for Studying a Corpus of Doctor-Patient Interaction, in Using Corpora for Language Research, edited by Jenny Thomas and Mick Short, Harlow: Longman, 92-109, 1996.

[Tian2001] Q. Tian, N. Sebe, M. Lew, E. Loupias and T. Huang, Content-based image retrieval using wavelet-based salient points, in Journal of Electronic Imaging, 2001.

[Tonnin2004] F. Tonnin and P. Gros, Interest point detection in wavelet and curvelet domains, in Proc. of Conf. on Photo-Optical Instrumentation Engineers , 2004.

[Torralba2004] A. Torralba, K. Murphy and W.T. Freeman, Sharing features: efficient boosting procedures for multiclass object detection, in Proc. of IEEE CVPR., 2004.

[Toutanova2003] K. Toutanova, D. Klein, C. Manning, and Y. Singer, Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, in Proc. of HLT-NAACL, 252-259, 2003.

[Tuytelaars2000] T. Tuytelaars and L. Van Gool, Wide baseline stereo matching based on local, affinely invariant regions, in Proc. of BMVC, 2000.

[Tuytelaars2004] T. Tuytelaars and L. Van Gool, Matching Widely Separated View Based on Affine Invariant Regions, in International Journal of Computer Vision, 59(1), 2004.

[Varma2002] M. Varma and A. Zisserman, Classifying images of materials: Achieving viewpoint and illumination independence, in Proc. of the European Conference on Computer Vision, 2002.

[VeriLook2007] VeriLook Face Detection, http://www.neurotechnologija.com/, June 2007.

[Vertan2000] C. Vertan and N. Boujemaa, Upgrading color distributions for image retrieval: can we do better ?, in Int. Conf. on Visual Information Systems,2000.

[Vidal2003] M. Vidal-Naquet and S. Ullman, Object recognition with informative features and linear classification. In Proc. of ICCV, 2003.

[Voorhees1993] E. Voorhees, Using WordNet to Disambiguate Word Senses for Text Retrieval, in ACM SIGIR, 171-180, 1993.

[Wallraven2003] C. Wallraven, B. Caputo and A. Graf, Recognition with Local Features: the Kernel Recipe, in Proc. of IEEE Int. Conf. on Computer Vision, 2003.

[Wang1998] J. Wang, G. Wiederhold, O. Firschein and S. Wei, Content-based image indexing and searching using daubechies' wavelets, in Int. Journal on Digital Libraries, 1(4), 311–328, 1998.

[Wang2000] Y. Wang, Z. Liu, and J. Huang, Multimedia content analysis using both audio and visual clues, in IEEE Signal Processing Magazine, 17(6), 12-36, 2000.

[Wang2005] X. Wang and J. Carroll, WSD using sense examples automatically acquired from a second language, in Proc. of HLT/EMNLP, 2005.

[Wang2006a] G. Wang, Y. Zang and Li Fei-Fei, Using dependent regions for object categorization in a generative framework, in Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition, 2, 2006.

[Wang2006b] X. Wang and D. Martinez, Word Sense Disambiguation using Automatically Translated Sense Examples, in Proc. of EACL Workshop on Cross-Language Knowledge Induction, 547-554, 2006.

[Weber1998] R. Weber, H-J. Schek and S. Blot, A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces, in Proc. of Int. Conf. on Very Large Data Bases, 1998.

[Weber2000] R. Weber and K. Böhm, Trading Quality for Time with Nearest-Neighbor Search, in Proc. of Int. Conf. on Extending Database Technology, 2000.

[Wechsler1998] M. Wechsler, E. Munteanu and P. Schäuble, New techniques for open-vocabulary spoken document retrieval, in Proc. of Int. ACM SIGIR conference on Research and development in information retrieval, 20-27, 1998.

[White1996] D. A. White and R. Jain, Similarity Indexing with the SS-tree, in Proc. of IDCE, 1996.

[Wilks1973] Y. Wilks, Preference Semantics, in Technical Report, Stanford University California, Department of Computer Science, 1973.

[Wilks1996] Y. Wilks and M. Stevenson, The Grammar of Sense: Is Word-Sense Tagging Much More than Part-of-Speech Tagging?, in Technical Report CS-96-05, University of Sheffield, 1996.

[Wilpon90] J.G. Wilpon, L.R. Rabiner, C.H. Lee and E.R. Goldman, Automatic recognition of keywords in unconstrained speech using hidden Markov models, in IEEE Transactions on Acoustics, Speech, and Signal Processing, 38(11), 1870-1878, 1990.

[Witkin1983] A. Witkin, Scale-space filtering, in Int. Joint Conf. on Artificial Intelligence and Computer Vision, 1983.

[Woodland1998] P.C. Woodland, T. Hain, S.E. Johnson, T.R. Niesler, A. Tuerk and S.J. Young, Experiments in Broadcast News Transcription, in Proc. of ICASSP, 909-912, 1998.

[Woodland2000] P. Woodland and D. Povey, Large Scale Discriminative Training for Speech Recognition, in Proc. of ISCA ITRW Automatic Speech Recognition, 7-16, 2000.

[Xia2004] C. Xia, J. Lu, B.C. Ooi and J. Hu, GORDER: an efficient method for KNN join processing, in Proc. of Int. Conf. on very large data bases, 756-767, 2004.

[Yahaoui2006] I. Yahiaoui, N. Herve, and N. Boujemaa, Shape-based image retrieval in botanical collections, in Pacific Rim Conference on Multimedia, 357–364, 2006.

[Yang1994] G. Yang and T.S. Huang, Human Face Detection in Complex Background, in Pattern Recognition, 27, 1994.

[Yang1999] M-H. Yang, D. Roth and N. Ahuja, A snow-based face detector, in Advances in Neural Information Processing Systems, 12, 1999.

[Yang2002] M.-H. Yang, D. Kriegman and N. Ahuja, Detecting Faces in Images: A Survey, in IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1), 34-58, 2002.

[Yang2004a] M-H. Yang, Recent Advances in Face Detection, in Proc. of ICPR, 2004.

[Yang2004b] Z. Yang, W.T. Ooi and Q. Sun, Hierarchical, Non-Uniform Locality Sensitive Hashing and Its Application to Video Identification, in Proc. of ICME, 2004.

[Yang2006] C. Yang, M. Dong and F. Fotouhi, Region based image annotation through multiple instance learning, in Proc. of ACM Multimedia, 2005.

[Yarowsky1995] D. Yarowsky, Unsupervised word sense disambiguation rivalling supervised methods, in Proc. of ACL , 189-196, 1995.

[Yavlinsky2005] A Yavlinsky, E Schofield, and S. M Ruger, Automated image annotation using global features and robust non parametric density estimation, in Proc. of Int. Conf. on Image and Video Retrieval, 2005.

[Yianilos1993] P.N. Yianilos, Data structures and algorithms for nearest neighbor search in general metric space, in Proc. of ACM-SIAM Symposium on Discrete Algorithms, 311-321, 1993.

[Zabih1994] R. Zabih and J. Woodfill, Non-parametric local transforms for computing visual correspondence, in Proc. of European conference on Computer Vision, 151-158, 1994.

[Zezula2003] P. Zezula, V. Dohnal, C. Genarro and P. Savino, Similarity Join in Metric Spaces, in Proc. of the Eur. Conf. on Information Retrieval, 2003.

[Zezula2004] P. Zezula, P. Savino, G. Amato and F. Rabitti, Approximate similarity retrieval with M-trees, in VLDB Journal, 2004.

[Zezula2007] P. Zezula, G. Amato and V. Dohnal, Similarity Search: The Metric Space Approach, in ACM SAC Conference Tutorial, 2007.

[Zhang1996] T. Zhang, R. Ramakrishnan and M. Livny, BIRCH: an efficient data clustering method for very large databases, in Proc. of ACM SIGMOD International Conference, 103–114, 1996.

[Zhang2004] R. Zhang, B. C. Ooi and K.-L. Tan, Making the pyramid technique robust to query types and workloads, in Proc. of Int. Conf. on Data Engineering, 313–324, 2004.

[Zhang2005] D.Q. Zhang, Statistical Part-Based Models: Theory and Applications in Image Similarity, Object Detection and Region Labeling, PhD Thesis, Graduate School of Arts and Sciences, Columbia University, 2005.

[Zhang2006] D-Q. Zhang, S-F. Chang, A Generative-Discriminative Hybrid Method for Multi-View Object Detection, in Proc. of CVPR, 2017-2024, 2006.

[Zhang2006] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid, Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study, in Proc. of Conf. on Computer Vision and Pattern Recognition Workshop, 2006.

[Zweig2006] G. Zweig, J. Makhoul and A. Solcke, Introduction to the Special Section on Rich Transcription, in IEEE Transactions on Audio, Speech and Language Processing, 14(5), 1490-1491, 2006.

7 Conclusion

This deliverable overviewed the state of the art in mono-media content indexing and retrieval. A wide range of literature has been reviewed and analysed in depth in order to guide the specification of the modules of VITALAS prototype V1 and to justify further research effort within WP2. We will not summarize here the conclusions of the different parts since they are quite independent and already synthesized. We let the reader refer to the conclusions and recommendations of each section (see table of contents at the beginning of this document). We just precise that the recommendations provided in this document are not specifications of future VITALAS system. The specifications of the prototype and final versions of the system will be addressed in further deliverables and milestones. Some technical projections of the use-cases could for instance not coincide with the technological recommendations of this document.