

Indexing, Browsing and Searching of Digital Video and Digital Audio Information

Alan F. Smeaton

School of Computer Applications, Dublin City University, Glasnevin, Dublin 9, IRELAND.
Alan.Smeaton@dcu.ie

In this chapter we examine various techniques for providing content access to information stored in a continuous medium, namely digital audio and digital video. Our coverage of audio is centered around post-processing the output of automatic recognition of speech or of phones and we describe the various approaches than have been taken in this area. In order to give reasonable coverage of the possibilities and limitations of content-based access to digital video information we sketch out at a high level, the approaches taken in various video compression algorithms, principally the MPEG family. We then address approaches to shot and scene boundary detection, choosing representative frames for browsing and for search, and various browsing interfaces that have been developed. We finish with an overview of the likely developments in this area in the future.

1. Introduction.

Having information available in digital format has many clear advantages including fast and cheap transfer of information, free and unlimited ridership, easy versioning and the capacity for easy and cheap storage of large amounts of information. Effective and fast content based access to information is something that has not been easy with analogue information but since practically all of the information we use in our daily lives is in digital format, or soon will be, content-based operations are an important goal. This will open up the prospects of content based operations such as search, filtering, alerting, summarisation and so on, which could not have been achieved heretofore. In this chapter we concentrate on content based operations on information stored in continuous media such as audio and video. Although the techniques we outline can be applied to other content-based operations besides retrieval we restrict our discussion to the searching task which we refer to as information retrieval, though we mention other applications in the conclusion.

Ideally, content based retrieval of any kind of information would be retrieval based on an understanding of the semantics of the objects in a collection. For any object, there are 2 approaches to the representation of content:

1. A human intermediary can interpret an object and generate keywords, captions, or some other kind of content description but the disadvantages of this approach are many and include at least:

- No consistency of interpretation by a single person over time;
- No consistency of interpretation among a population of interpreters;
- No universally agreed format of the representation, whether keywords, captions or some knowledge-based formalism;
- cost !

We can see evidence of the first three of these reasons in the difficulties we have in the human classification of books, for example in the ACM Computing categories, or in the difficulties we have in classifying web pages in Yahoo! or the Open Directory Project [1].

2. The second approach to representing content is to have an automatic interpretation or transcription of objects by computer. The advantage of this is that it leads to a consistent interpretation of meaning, even if it is low-level and often wrong !

These general principles apply to information retrieval on all kinds of objects including text and other media and elsewhere in this volume other chapter authors discuss various aspects of this. Here, and specifically in the context of audio and video information, we can identify the same two general approaches where user-assigned descriptions of audio or video has more or less the same difficulties and approaches as user-assigned descriptions of text, so that isn't of much interest to us here. What is of interest to us here is the ways in which we can do automatic indexing and then information retrieval, on audio and video.

The current trend in content-based retrieval systems for non-text is based on 3 key ideas:

- A) Successful content-based retrieval systems are domain-specific and only work in those domains though they may be ported elsewhere;
- B) Automatic understanding tools have been (to date) impossible to develop and so must be replaced by interactive ones which involve users;
- C) Humans should be given primitive tasks that can be performed consistently rather than complex ones which yield variable outputs;

These principles in general point to a somewhat bleak picture vis-à-vis how we can do fully automatic indexing and retrieval of audio and of video but in fact quite a lot of progress has been made in these areas in recent years. Before examining what that progress is, and where we currently stand there are some further characteristics of multimedia objects, and of video in particular, which we will state here in order to give a full picture of the difficulties and challenges of information retrieval on these media. These are:

- ◇ Multimedia objects such as video clips have multiple dimensions and how we view an object, what our task is, what we are looking for, and so on, will all elicit different properties. Ideally, to cater for the many different interpretations of, say, a video, from many potential searchers, we would like to be able to capture all these possible features, but we cannot.

- ◇ We may eventually require retrieval based on a set of properties or types not initially captured by the system at indexing time, so our system should ideally be extensible in its index representation of multimedia objects to allow it to go back and "re-index" or perhaps index at query time.
- ◇ We should develop suites of retrieval techniques that can be used for sub-groups of features rather than developing a single retrieval technique which operates over the entire set of properties of a multimedia object. Ideally each technique should operate on the principle of an inexact match between an information need and an object, and should be based on an overall object ranking. Furthermore, we should be able to integrate these sets of ranked lists into a single overall ranked list combining individual evidences for each object into an overall Retrieval Status Value (RSV) given that each object could will involve more than one group of features.
- ◇ We must understand and allow for the fact that the automatic interpretation of objects that we handle will be both incomplete (with some parts of the description missing), inexact (some part(s) of the description may have certainty values associated) and possibly even erroneous (the interpretation probably involves automatic processes, and even human-assisted will have errors).
- ◇ We must also understand that query specifications will also be incomplete and may be refined as with document retrieval but in document retrieval, a document has many but a still definable number of interpretations, i.e. answers to questions. A video is even more content-rich compared to a text document and thus answers many and very diverse queries.

All these factors contribute to information retrieval on video and on audio being very difficult with almost everything stacked up against us. In fact what has happened in the area is that to date we don't have effective retrieval in the way we have come to know for text-based retrieval with the cycle of formulate and submit query, system generates ranking, browse ranked list, locate relevant document, click and read. Instead we have more of a browse-query-browse interaction with the requirement for browsing coming from the fact that one cannot easily get a *gist* of a video or audio clip compared to assessing the contents of a text document. This means that the whole concept of information retrieval on audio and on video becomes very different compared to information retrieval on text documents. We shall return to this point later on.

Before moving on to describe how current techniques for information retrieval on audio and video operate there is one other important element of the context in which we can do content operations on such media of which the reader should be aware. The huge strides that the computing industry has made over the last decade or so, to put digital multimedia information on our desktops and in our homes have been achieved by a concentrated effort into developing technologies to capture or create, store, transmit, render and display this media. Chief among this has been the development of encoding and compression formats for media which have the over-riding constraint

of achieving maximum compression for minimum quality loss in order to make storing and transmitting the media possible. We thus have a situation where video and audio information is stored in a digital format which has been developed without any consideration given to how that information might be manipulated by content. The "engineering" aspect of delivering digital multimedia has been virtually the only concern in developing these encoding standards and formats, and that has started to change only very recently. When we try to do information retrieval on encoded audio and video we find that we are almost fighting against the format in which it is encoded and at the very least, unless we have huge computation resources to decode everything back to raw bits, we are constrained to leveraging whatever we can from the encoding, which as we know was driven by considerations of compression. This makes our task even more difficult with compression formats having a huge impact and limitation on the possibility for content based operations.

The organisation of this chapter proceeds as follows. In the next section we examine information retrieval on digital audio, concentrating on retrieval from spoken audio. Following that, and the main part of the chapter, we look at retrieval and browsing of digital video. Part of this section gives a thumbnail sketch of the approaches taken to video compression before we move on to cover indexing, browsing and content-based operations. In the final section we examine more general questions about retrieval aspects of audio and video.

2. Information Retrieval on Digital Audio.

When we refer to digital audio information we refer to audio recordings of speech, music or other sounds. The "other sounds" category is generally restricted to specialist applications (bird sounds, whale whistles, etc.) and to sound effects (games, movies) so we are left with speech and with music.

There are several good textbooks which address issues such as psychoacoustics, the human hearing range, loudness, and the perception of sound [2,3]. For our purposes here all we need to know is that sound is a continuous vibration which is sampled at a given rate which leads to a quantization of the analog waveform into digital format. A higher sampling rate means less quantization noise which means better quality. Audio CD recordings are sampled at a rate of 44 kHz and each of the samples is stored at 16 bits, for each (of 2) channels giving stereo reproduction. Lower sampling rates such as telephone quality audio use up less space in digital form, and require less bandwidth to store and transmit files, but sound poorer, so there is a clear tradeoff.

Once audio has been digitised in this way there are literally scores of formats in which it can be represented. The WAV form is common and the samples can be 8- or 16-bit, and there are many sampling rates that can be used, but there is no compression so it is raw, and uncompressed format. Other formats achieve large compression ratios and include AU, VOX, RealAudio, TSP, VMF, AIFF and so on. MP3 is a format that has become hugely popular on the internet for encoding music and it does this by using higher compression at the parts of the audio spectrum where

human hearing is at its least discerning, an approach referred to as perceptual compression.

Another audio encoding format worth noting is MIDI, an international standard for digital music which has high acceptance in the music community. In MIDI, "sounds" are encoded as one or many streams with each stream recording a specification for each of the notes, duration, volume, etc, and also the type of the sound being played, so a MIDI file is actually an ASCII text file. For example, the code 20 refers to a church organ while 117 is a taiko drum and 124 is a bird tweet ! [3, pp93].

Once audio information has been digitised we can explore the possibilities for content-based access. For non-speech information the limits are a combination of our imagination and our technology. If information is stored in MIDI format then there are several approaches to finding tunes from a MIDI song database using information retrieval techniques derived from text-based IR [4]. With the explosive growth in the use of MP3 as an encoding format there are huge possibilities for content-based indexing and retrieval systems but at the present time MP3 files are accessed almost exclusively through their metadata (title, artist, etc.) rather than directly on their content, though this is an active area of research.

For audio encoding of speech, the situation is more advanced. Speech processing has always been an target of artificial intelligence and [5] is a good, short, snappy, complete and thorough survey of the problems and solutions to speech processing. Although that article is perhaps a bit dated and there has been some progress since then, speech processing is still short of human capabilities but by limiting the domain it allows it to be productive and there are commercial developments. There is an acceptance that speech will not replace the mature, established and efficient alternative for data input (keyboard/mouse) but can be combined with it. When it comes to automatic processing of spoken audio collections then the applications which are attracting most attention are in radio/TV news retrieval, searching archival radio/news broadcasts, video and audio email, searching audio archives of meetings, lectures, etc.

Two utterances of the same words by the same person under the same conditions generate very different waveforms because of the variability of human speech generation, air temperature, acoustics, etc. Given that digital audio is a waveform sampled at a given frequency and into a given bit-size (8- or 16- bits), possibly with lossy compression added, it follows that direct wave-to-wave matching of audio will not yield any kind of reasonable performance. Variations due to loudness, pitch, brightness, bandwidth, harmonicity, and others are all continuous variables and are equivalent to colour and texture in images. Thus all speech document retrieval systems are thus based on some kind of recognition of spoken words or of phones.

A spoken word is exactly the same as a written word, albeit with the difficulties of determining the boundaries between words since we tend to speak continuously and without word breaks. A phone is a sub-word unit, equivalent to a unit of pronunciation, larger than a letter, but smaller than a word. For example, the phrase "more details" consists of the 9 phones:

m o o r d i i t e i l z

with double letters used in the alphabet to represent some single phones. These phones are taken from one of several phone alphabets used commonly, with no real agreement on a standard or on which alphabet is best.

Once phones have been identified then a speech recognition system will try to group these together into words. This is a non-trivial task since there are no word bounds and since phone recognition usually outputs a lattice of phones rather than a single stream. This means that for many possible phones, a phone recogniser will output more than one candidate phone, with associated probabilities of likelihood. Phone recognition and word identification work best when they are collaborative processes, re-enforcing each other with certain phone combinations giving words which are not in a dictionary or unlikely combinations, and commonly occurring words strengthening the chances of certain phone combinations actually occurring.

Some approaches to information retrieval on speech have been based on full word recognition techniques while others have been based on indexing spoken text by the phones. These approaches can be roughly grouped as follows.

Approach 1: Word Spotting

Instead of trying to do recognition of full speech, the Cambridge/Olivetti VMR (video mail retrieval) project [6] did word-spotting, i.e. given a pre-defined vocabulary of the order of some tens of words, process the spoken audio component of video mail looking for these words and these words only, and use them as indexing terms. By this restriction to a reduced set of key words the problem of speech recognition is reduced in complexity and becomes manageable. A user's query in this system is to search for these keywords and the keywords chosen are good discriminators between messages in a VMR application.

The reason why this works is because the speech recognition can perform effectively and this is because it has a limited vocabulary for recognition. If a word in the stream cannot be recognised then it moves on assuming it is not one of the keywords.

Approach 2: Speaker Recognition

The Jabber project at the University of Waterloo [7] applied speaker-independent continuous speech recognition to the audio recording of a meeting but like all such attempts at such recognition has had problems because it tried to recognise all of the words spoken by everyone. However, one of the spin-off benefits of this approach is that it can recognise **who** has done the speaking at any time and a visual summary of the dialogue turn-taking can provide a kind of navigational support through meeting archives if speaker recognition can be done accurately enough.

This shows that it is possible to leverage reasonably effective retrieval from even moderate processing of the audio signal. Both the VMR and Jabber projects used off-the-shelf or tailored recognisers but even these speech recognition systems need quite an amount of training before they can be effective.

Approach 3: Phone-based Retrieval

Instead of recognising word-level tokens in speech recognition, an alternative is to recognise sub-word units, namely *phones*. A project at ETH-Zürich [8] indexed (German) radio news broadcasts by phones, accurately, into a lattice, so for each

phone it had a set of possible candidates with probabilities. For these sets of candidates phones there are paths through the phone choices which correspond to words. At Dublin City University our own Taiscéalaí system [9, 10] took the same approach and was engineered into a real operational system. The Taiscéalaí system captured RTE Radio 1 news broadcasts twice daily, recognised phones, broke the stream of phones into overlapping windows, and generated a document for each window which was indexed as a retrievable unit of information. User queries (text) were turned into phones via a dictionary lookup and the matching and retrieval operation was based on a bag of tri-phones from the query matched against bags of triphones for indexed windows of audio broadcasts. Each window was 30s in length with a 10s overlap.

The Taiscéalaí system gives us our first glimpse of the difficulty of information retrieval where the objects being searched are continuous streams of information, rather than discrete objects such as text documents. If we assume that we are seeking a ranked list from an information retrieval operation then such an operation on a collection or library of such objects has two dimensions; the first is related to finding which how the objects are ranked and the second is related to the position or offset within each object, where the user is pointed to. A system which retrieves 30 minute video or audio clips and presents a ranked list of them to a user without any browsing or navigation within the objects is not of much assistance. In Taiscéalaí we computed a measure of similarity for each 30s window and we then aggregated these into scores for regions of each broadcast and we presented, visually, the summary for each broadcast as shown in Figure 1. This screendump shows the 6th, 7th and 8th ranked broadcasts returned from the query "*head of European bank France Germany Kohl*". In the broadcast of 03 May 1998 there is a region between about 1:30 and 2:30 which the timeline shows seems to have several query term occurrences and the user has selected a short fragment of this broadcast around the 2:20 mark to be played. The vertical peaks on these timelines represent short pauses and the different shadings on the x-axis correspond to different broadcast sources, i.e. in-studio, telephone interview, outside broadcast, music, etc.

The problem of retrieval of (multimedia) objects and retrieval or navigation within these objects is analogous to passage retrieval in classical text-based information but is more complicated because it is more difficult to get an overview of an audio or video clip than a text document. This is a problem that will arise again when we look at information retrieval from digital video in the next section.

Approach 4: Word-based Audio Retrieval

Despite the fact that speaker independent continuous speech recognition is not 100% accurate and is computationally demanding there are several examples of work reported where speech recognition is applied to spoken audio to support subsequent information retrieval. The biggest catalyst for this has been the effect of the TREC spoken document track. TREC is an annual benchmarking exercise carried out since 1990 in the area of information retrieval. Coordinated by the National Institute for Standards and Technology (NIST) in the US, NIST is a world-wide operation which has involved most of the major information retrieval research groups in industry and academia. TREC started as an initiative addressing text-based information retrieval only but then merged into almost a dozen "tracks" or variations including retrieval in

different languages, cross-lingual IR, retrieval from documents corrupted from an OCR process, interactive retrieval, information filtering and information retrieval from spoken documents. In this latter track, the audio from some hundreds of hours of news broadcasts was the collection and the task was to find broadcasts which were relevant to a given query. The logistics of the operation of TREC are best described in [11].

In the spoken document track which has been ongoing for three years, many of the participating groups take the approach of training a speech recogniser to recognise spoken words and they follow this with some variation of conventional text-based information retrieval on the recognised output. Much of this work is based on taking

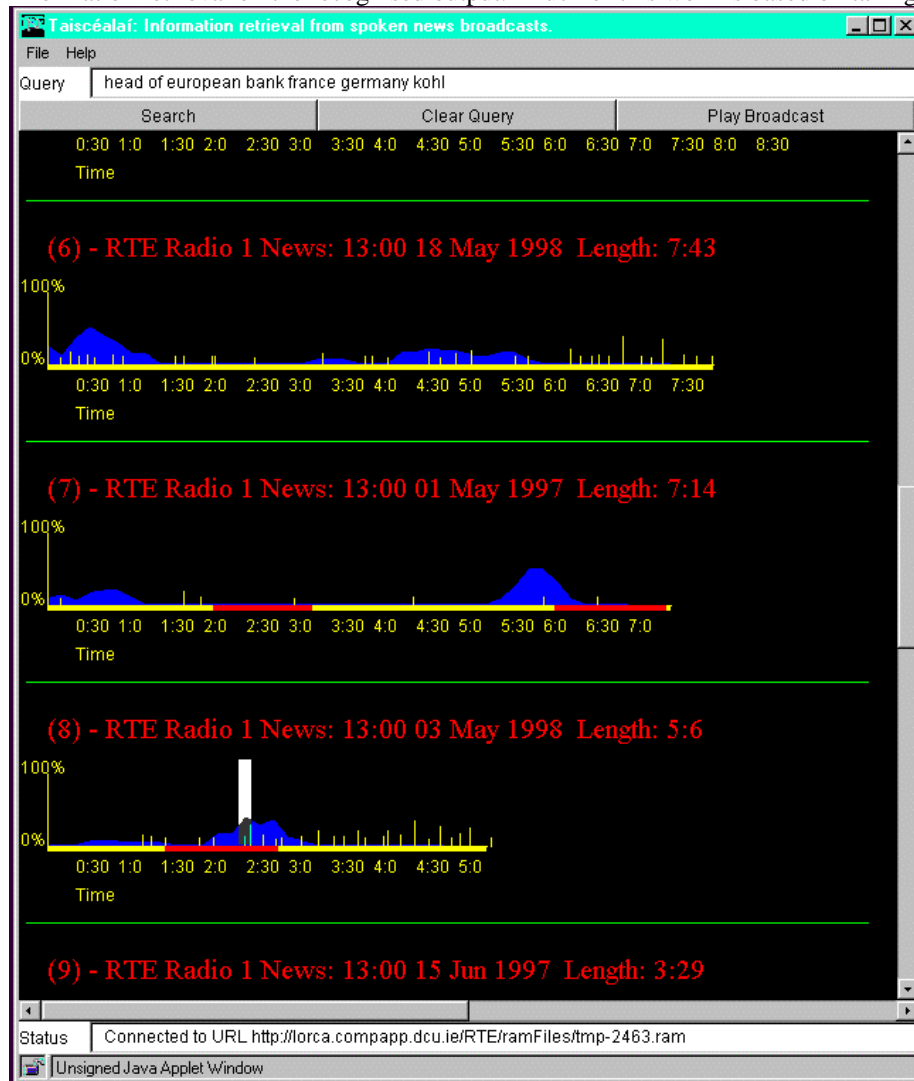


Fig. 1. Screenshot from the Taiscealaí System

account of and making allowances for the occurrence of the kind of speech recognition errors typically encountered in speech. Examples of systems which take this approach can be found in [12].

3. Information Retrieval from Digital Video

Video is basically a sequence of images relayed at a constant speed, normally 25 to 30 frames per second, with a synchronised audio track. Chapter 5 of [3], has details of analog video fundamentals such as aspect ratio, sync, horizontal and vertical resolutions, frame rates for motion (25 to 30), colour fundamentals (RGB), colour video and TV, video formats and worldwide TV standards such as NTSC in the US, PAL in most of Europe, and SECAM in France. It also covers video performance measurements, colour test cards and video recording equipment. Here we are concerned only with the digital encoding of video.

To display a single image of TV quality video requires 720 Kbytes and frames must be displayed at at least 25 frames per second to get the effect of smooth motion. This means that 18 Mbytes of storage is required for each second of a video without compression and for a 90 minute film this would be almost 100 Gbytes of storage. It also has the alarming implication that a CD-ROM with a storage capacity of 648 Mb and a data transfer rate of 150 Kb-per-second (the data transfer rate for original CD-Roms) would only be able to store 36 seconds of video, and it would take almost five seconds to download and display each frame.

Clearly the display and manipulation of TV quality video on computer screens has two technical barriers namely storage capacity and data transfer rate and these can be solved by data compression techniques and improved hardware and software. When it comes to video compression the formats that matter are the various MPEGs, H.261/p*64 and DVI, but really the most important is the MPEG family.

A fundamental part of all video compression approaches is motion compensation, identifying motion in adjacent video frames and spotting and transmitting only the differences between adjacent frames. Of course this does not apply to adjacent frames which straddle shot or scene changes. Determining differences between frames on the basis of pixel-to-pixel comparisons is too simplistic because much video content will have some kind of camera motion as cameras are never stationary and can pan, zoom, tilt, boom, and so on or be noisy, or have slight movements. Thus in video compression algorithms, frames are divided into blocks which are larger aggregates than pixels and motion compensation is tested between the blocks. In this way, slight noisy movements of the camera can be incorporated without incurring loss of effective compression, as well as facilitating efficient encoding with the larger deliberate camera moves like panning and zooming.

Of the contemporary video encoding standards in use today it is the MPEG family which are the most important. What makes the MPEG standards attractive is that these are standards agreed upon by a large community with a broad representation,

ahead of the chaos that exists with, say, image formats. What is especially notable is that during the development of the MPEG standards nobody on the development group had existing proprietary video standards that they wanted to push, so the various actual MPEG standards put forward have been more computationally and conceptually complex than anything in place at the time of their development. The implementation of the MPEG standards followed on after their specification.

At present there are 4 MPEG standards whose specification is complete or nearly so. MPEG-1 has been around longest and encoding and decoding is achievable on desktop computers. MPEG-2 has the same general approach as MPEG-1 but delivers higher quality and is used in digital TV broadcasting. MPEG-2 encoding requires specialist hardware and even MPEG-2 decoding and playback is not yet commonplace on desktop computing requiring, as it does, special hardware. MPEG-4 encoding and playback is a technology which has not yet been developed to the extent that it can be used in anything but laboratory test environments while MPEG-7 is still at the draft specification stage.

MPEG-1 encoding turns a 3D video sequence (x-axis, y-axis and time) into a one-dimensional bit stream for transmission [13]. MPEG-1 uses a frame size of 352x288 pixels at 25 fps giving VHS quality at a fixed rate of 1.5 Mb/sec or just under (which is the data rate from a CD-Rom), though larger frame sizes and different FPS rates can be encoded. MPEG decoders and players are common and available and on workstation or PC can decode in real-time for 25 FPS, but few encoders are available without hardware add-ons. Each frame is compressed breaking it into 8x8 pixel blocks for inter-frame and 16x16 pixel macroblocks for intra-frame motion compensation. Macroblocks are strung together to form slices which are combined into a picture. A number of pictures are grouped together into a group of pictures (GOP) to form a random access unit to allow forward/rewind with no dependencies between GOPs and hence handling of breaks in transmission.

In MPEG-1 there are 3 types of frames namely:

1. I-frames are intracoded frames, meaning they are encoded block-by-block independently of adjacent frames as if they were still images, and they are encoded with lossy compression using JPEG compression.
2. P-frames are forward-predicted frames, encoded with reference to the most recent previous I- or P-frame with motion-estimation and macroblocks vector-matched.
3. B-frames are bidirectional predicted frames coded with reference to previous and following I- or P-frames with motion-estimation and encoding similar to P-frames.

The pattern for I-, B- and P-frames will vary from encoder to another but could be something like the following

I - B - B - B - B - P - B - B - B - B - P - B - B - B - B - I - ...

A GOP is a frame pattern of I-, B- and P-frames generating a bit stream which is further compressed using Huffman coding, which yields great compression and reduces the whole video stream, including audio, to about 1.5 Mbits per second.

MPEG-1 was initially targeted at multimedia applications reading from CD-Rom but also supports frame based random access through the video, FF/Rew and reverse playback.

Beyond MPEG-1 there is MPEG-2 which has data rates of between 2 and 9.8 Mbits per sec, enough for high definition TV. MPEG-2 is 720x576 pixels and is used for the transmission of digital TV and video on DVD. It is a superset of MPEG-1 and takes the same approaches to encoding as MPEG-1

There was an MPEG-3 slated to cater for HDTV but MPEG-2 proved adequate for these requirements so the MPEG-3 standard was dropped as work on the specification for MPEG-4 had already been started. MPEG-4 has been recently finalised and is targeted at very low bit rate encoding of audio-visual interactions requiring a completely new approach to encoding based on human-computer interactions. Part of this involves identifying objects in a frame as coloured shapes and tracking these objects from frame-to-frame and applying shape compression which is very effective, all without knowing what the shapes actually represent. Instead of being block-based as MPEG-1 and -2 are, MPEG-4 is based on object compression and represents video as a series of planes, superimposed upon each other to give the final rendered picture. This will allow future multimedia applications with extended interactive functionalities and access to actual content, i.e. the objects in the video, where the rendering of the frames can even be personalised in some way. This encoding of objects allows deconstruction and reconstruction in an object layer but the identification of objects is the biggest challenge here; tracking and compression of objects is currently achievable for synthetic or artificial video such as animated cartoons but not for video of natural scenes of objects. While the specification of MPEG-4 is complete and available, the implementation of this has not yet been achieved fully.

The final MPEG standard is MPEG-7 which is unusual in the MPEG family as it has visual and audio elements but also it has a content descriptor stream where the semantic content of the video is encoded and represented. Clearly this is targeted at content based operations such as search but as this specification is still under debate we have only speculation as to how it might end up. It is believed though that the descriptor stream will be some markup language similar to, if not part of, XML.

We now concentrate on indexing and retrieval from the digital video stream, but we note that as video is a continuous media, usually combined with an audio track and for most effective access an application should use both synchronised media for retrieval.

It is relatively straightforward to treat digital video as a binary blob and for each video to store aspects like date, title, director plus a textual description of video contents and to do little more than search the metadata associated with a video, or the video transcript. This isn't video retrieval though and we won't discuss this further.

To see how video retrieval can work we need to examine what exactly a video is and how it has been encoded. Video is a sequence of individual shots of variable lengths butted together in some way, and played as a continuous stream into a 2D window. Thus it has 3 dimensions: x , y and t .

The way to make progress in manipulating video content is to structure the video in some way and identify the shots and then segment the video into a list of shots using automatic shot boundary detection (SBD). This task of automatic video segmentation is quite difficult as production level video incorporates such tricks as fade-in and fade-out, dissolving, morphing, wipes and many other chromatic effects and these are surprisingly commonplace in TV and in movies. Programmes such as gardening

programmes, cookery programs, TV adverts, etc. all incorporate such effects, even live coverage of sports with action replays as on-the-fly post-production effects.

Initial attempts at video segmentation were based on processing the primitives in a video which are similar to the primitives in a single-frame image, but with time added, namely colour and associated histograms and their within-frame distribution, texture, intensity/brightness, etc. These can be used to detect scene changes by measuring shifts in the overall histogram or in the distribution of colours within a frame. Essentially they are based on measuring inter-frame distances and are really an application of still image retrieval [14], but they can be confounded by commonplace camera techniques such as zooming and panning, booming, tilting and tracking, or a combination of all of these. They are even more thrown by fade-in and gradual scene changing.

In our work at Dublin City University we evaluated the effectiveness of several approaches to SBD as follows:

- Measuring inter-frame difference based on differences between colour histograms for the entire frame;
- Calculating the edges around the objects in adjacent frames using Sobel filtering and measuring the differences between these;
- Measuring colour moments between adjacent frames and using this to determine shot bounds;
- Extracting motion vectors from MPEG-1 encoding and using these to indicate shot bounds;

To evaluate these we took 8 hours of broadcast TV including many different program types like gardening programmes, news, soap operas, detective series, a comedy programme with TV commercials interspersed, and we manually marked up all shot bounds. From these 720,000 frames of video we identified 5380 hard shot cuts and another 779 fades or dissolves and we used this as a ground truth against which to compare the different techniques. We found that all techniques perform at about the same level with both recall and precision at around 85% to 90%. As a refinement of this, instead of setting a fixed threshold we developed a method of dynamically adjusting the threshold depending on the programme genre, as determined by measuring the visual "noise" in a programme [15]. We also experimented with running all four SBD methods and combining the outputs into one decision [16]. Our conclusion from this work is that the more sophisticated approaches do give some improvement over the basic colour histogram method but not enough to merit the computational cost required. On our hardware configuration, SBD using colour histograms takes about the same time as the length of the video being analysed but using edge detection or colour moments takes twice as long. On balance, colour histogram based detection gives adequate performance.

Once a video stream has been segmented into shots we are then faced with the problem of determining how can it be indexed or browsed or viewed. It is essential to present video visually because it is a visual artifact in the first place. By segmenting a video into shots we reduce the problem to one of indexing a series of single scenes or shots. The common approach to this is to find a representative frame from each shot, above a certain length, and to present a video as a series of images, a kind of visual storyboard. A variety of approaches have been taken to choosing the representative frame for a shot. The simplest was to choose the frame in the middle of a shot but

this may not be a good choice to essentially randomly pick a frame. Another approach is to choose the first frame, but in the case of a shot bound which is not a hard cut this can yield a frame which is half one shot, half the previous one. In our work on the *Fischlár* system [17] our approach is based on treating the clip as a set of individual images and the "image" or frame with the most average colour histogram or the one which is most similar to the others (using image matching techniques) is chosen. Thus the video stream is now a list of images where each image has an associated "length" of the clip from which it was taken. This approach, though not the details of our technique, is common in video indexing systems.

Once video is structured in some way we can support a visual search on keyframes from shots, but the most natural way to navigate through video is not to use search but to use browsing. The difficulty with this though is that browsing keyframes is still browsing through a very large information entity for long video streams perhaps of the order of hours, so simply structuring video into shots does not achieve much on its own. If we take an archive of many days of video content, as we have in our *Fischlár* system [17], then browsing through keyframes is adequate only for browsing within a single programme. This must be complimented by using programme metadata such as title, actors, programme genre, date, time, TV station or whatever, to select the programmes to be browsed. This returns us to the point made earlier about information retrieval through video requiring more of a search-browse interaction than for text-based information retrieval.

Once we have narrowed the "library" or collection of video to a unit of the order of size of a programme, the task then becomes browsing within this programme in order to find segments of interest. In [18] we present a framework for the development of browsing interfaces to video. Two of the most interesting ones which illustrate different approaches are shown in Figures 2 and 3

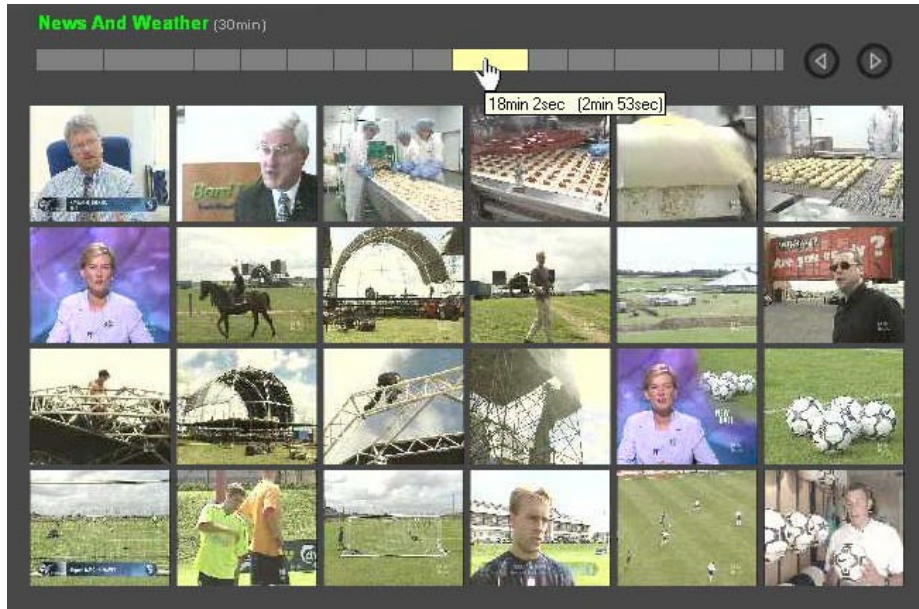


Fig. 2. Screenshot from Físchlár showing keyframes taken from the middle of a 30 minute News and Weather broadcast.

In Figure 2 we see 24 keyframes, each representing a shot of length greater than 1s, with the whole programme represented as a timeline in the top part of the frame, and the region of the programme that the 24 keyframes represents at around 18 minutes 2 seconds, clearly highlighted and visible. In Figure 3 we see a hierarchical video browser with the top line of keyframes spanning the whole 30 minute broadcast and one of these (the 4th frame from the left on the top level) selected and drilled down to the second level which covers a smaller region of the broadcast and one of these keyframes in turn is also selected and expanded to show a sub-sub region of the video.

- segmenting the video stream based on colour histogram changes, shape and texture measurements between frames and allowing for the kind of scene transformations mentioned earlier; there is also camera motion detection for pan, zoom, tilt, etc. ;
- object detection looking specifically for faces and for text captions in the image and matching these against a database of known VIP faces in order to try to detect the presence of these VIPs;
- caption and text extraction from still frames in the videos which was then input into an OCR program;

All these tools operated on the video content and allowed the generation of video skims or summaries with transcripts, the presence of known VIPs, and so on, all presented. The retrieval tool in Informedia allows a user's text input query to be matched against the transcript (allowing for speech recognition errors) and against recognised captions to select segments of video which can be skimmed by viewing a series of representative frames with associated keywords (from the dialogue) and a user can choose a frame and run it as a query to find video clips like that one. Informedia is successful and has attracted attention because it was the first to integrate so many complimentary techniques for information management into the one system for managing video material.

4. Conclusions on Managing Digital Video and Digital Audio

In reading this chapter it should be clear that the element of browsing is an essential part of navigation through video and audio content, more so than with text or image information. This means that we cannot simply take text-based information retrieval and apply it to video but we must re-think the whole user-system interaction and integrate browsing and searching as seamlessly as possible. In practice we are only starting to do this as technology has up to now prevented us from doing so. Soon, with digital TV broadcast into our homes, the demand for control on this content will grow dramatically and as we presently stand we are not ready for these demands.

Most of the research systems developed to day have concentrated on desktop-based tools for managing video content. People managing digital TV, which will be the largest userbase and application for this work, do not sit at desktops or have mouse/keyboards for control as they lie on couches and use small remote control devices. The interfaces we develop for video content manipulation will have to reflect this and they do not to date.

Another area where there are big changes happening which will affect video content management is the unpredicted and enormous growth in mobile communications, especially in Europe. We soon will have GPRS telephony and early in 2002 we expect to have third generation mobile phones with enough bandwidth to stream MPEG-1 quality video to mobile handheld devices. This has the potential to completely change the way in which we watch TV and manage our TV viewing. Little work has been done to date, however, on video streaming and video content management on a PDA or mobile phone.

Finally, some pointers as to what the likely applications for video content management will be. Surprisingly, we would not expect to much need for classical information retrieval searching though people will want to do keyword or key phrase searching of archives of broadcast material. Applications which are based around push technology such as filtering, alerting and summarisation of video content will grow in importance, especially as the mobile communications market grows in size and importance. Users will want to have SMS or WAP alerts as to the status of the programmes they asked to be recorded and analysed. Users will want to be able to stream video, even of preview quality, to their mobile PDAs. Users will want to be able to access trailers or summaries of video material and users will want lots of personalisation in order to make the volume of accessible material, manageable. Some demand will exist for access to broadcast and archived audio material but this will be specialist and niche. The real market and the real scientific challenges remain in the area of video content indexing and navigation and there are still lots of problems to be solved.

References

1. <http://www.dmoz.org> visited 16 August 2000.
2. Tannenbaum, R. S.: Theoretical Foundations of Multimedia. W. H. Freeman and Company, The Computer Science Press, New York, (1998).
3. Koegel Buford, J.F.: Multimedia Systems. ACM Press, Addison-Wesley Publishers, New York (1994).
4. Downie, J.S. and Nelson, M.: Evaluation of a simple and effective music IR system. In: *Proceedings of the 22nd ACM-SIGIR Conference*, Athens, Greece, July 2000.
5. Rudnicky, A.I., Hauptmann, A.G. and Lee, K-F.; Survey of current speech technology. *Communications of the ACM*. 37(3), 52-57, (1994)
6. Jones, G.J.F., Foote, J.T., Spärck Jones, K. and Young, S.J.: Retrieving spoken documents by combining multiple index sources. In *Proceedings of SIGIR 96, Research and Development in Information Retrieval*, 30-38, Zürich, ACM Press, (1996).
7. Kazman, R. and Kominek, J.: Supporting the Retrieval Process in Multimedia Information Systems. In: *Proceedings of HICSS '97*, Vol. VI, 229-238, (1997).
8. Schäuble, P. *Multimedia Information Retrieval*. Kluwer Academic Publishers (1997).
9. Smeaton, A.F., Morony, M., Quinn G., and Scaife, R.: Taiscéalaí: Information Retrieval from an Archive of Spoken Radio News. in *Proceedings of the Second European Conference on Research and Advanced Technology for Digital Libraries (ECDL)*, Crete, C. Nikolaou and C. Stephanidis (Eds.) Springer LNCS 1513, 429-442, (1998).
10. Quinn, G. and Smeaton, A.F.: Optimal Parameters for Segmenting a Stream of Audio into Speech Documents", G. Quinn.: in *Proceedings of the ESCA ETRW Workshop on Accessing Information in Spoken Audio*: 19-20 April 1999, Cambridge, UK.
11. Voorhees, E.M. and Harman, D.H. The Sixth Text REtrieval Conference (TREC-6). *Information Processing and Management* 36(1), 3-35 (1999).
12. Garofolo, J., Voorhees, E., Auzanne, C., Stanford, C. and Lund, B. 1998 TREC-7 Spoken Document Retrieval Track Overview and Results. In *NIST Special Publication 500-242: The Seventh Text REtrieval Conference (TREC 7)* 79-90, (1999). (also available at http://trec.nist.gov/pubs/trec7/t7_proceedings.html last visited 10 August 2000.
13. Sikora, T. MPEG Digital Video-Coding Standards. *IEEE Signal Processing Magazine*, 82-99, (1997).
14. Eakins, J.P. *Retrieval of Still Images by Content*. This volume (2000).

15. Smeaton, A.F., Gilvarry, J., Gormley, G., Tobin, B., Marlow S. and Murphy, N. An Evaluation of Alternative Techniques for Automatic Detection of Shot Boundaries in Digital Video. In: *Proceedings of the Third Irish Machine Vision and Information Processing Conference*, Dublin, September 1999.
16. Browne, P., Smeaton, A.F., Murphy, N., O'Connor N., Marlow, S., Berrut, C. Evaluating and Combining Digital Video Shot Boundary Detection Algorithms. In *Proceedings of the Fourth Irish Machine Vision and Information Processing Conference*, Queens University Belfast, September 1999.
17. Lee, H., Smeaton AF., O'Toole, C., Murphy, N., Marlow S and O'Connor, N.E. The Físchlár Digital Video Recording, Analysis, and Browsing System. In *Proceedings of RIAO '2000: Content-Based Multimedia Information Access*, Paris, France, April 12-14, 2000
18. Lee, H., Smeaton, AF, Berrut, C., Murphy, N, Marlow, S. and O'Connor, N. Implementation and Analysis of Several Keyframe-based Browsing Interfaces to Digital Video. *To appear in Proceedings of the Fourth European Conference on Digital Libraries*, Lisbon, Portugal, September 2000.
19. <http://lorca.compapp.dcu.ie/Video/SLinksF.html> Link visited 13 August 2000.
20. Zhang, H., Low, C. and Smoliar, S.. Video Parsing and Browsing Using Compressed Data. *Multimedia Tools and Applications*. 1:89-111, (1995).
21. Perry, B., Chang, S-K, Dinsmore, J, Doermann, D, Rosenfeld, A and Stevens, S. Content-Based Access to Multimedia Information: From Technology Trends to State of the Art. Kluwer Academic Publishers, 69-77, 2000.