EURASIP Journal on Applied Signal Processing 2003:2, 209–222

Search the Audio, Browse the Video—A Generic Paradigm for Video Collections

Arnon Amir
IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA
Email: arnon@almaden.ibm.com

Savitha Srinivasan
IBM Almaden Research Center, 650 Harry Road, San Jose, CA 95120, USA
Email: savitha@almaden.ibm.com

Alon Efrat
Computer Science, University of Arizona, Tucson, AZ 85721-0077, USA
Email: alon@cs.arizona.edu

Received 18 April 2002

The amount of digital video being shot, captured, and stored is growing at a rate faster than ever before. The large amount of stored video is not penetrable without efficient video indexing, retrieval, and browsing technology. Most prior work in the field can be roughly categorized into two classes. One class is based on image processing techniques, often called content-based image and video retrieval, in which video frames are indexed and searched for visual content. The other class is based on spoken document retrieval, which relies on automatic speech recognition and text queries. Both approaches have major limitations. In the first approach, semantic queries pose a great challenge, while the second, speech-based approach, does not support efficient video browsing. This paper describes a system where speech is used for efficient searching and visual data for efficient browsing, a combination that takes advantage of both approaches. A fully automatic indexing and retrieval system has been developed and tested. Automated speech recognition and phonetic speech indexing support text-to-speech queries. New browsable views are generated from the original video. A special synchronized browser allows instantaneous, context-preserving switching from one view to another. The system was successfully used to produce searchable-browsable video proceedings for three local conferences.

Keywords and phrases: automatic video indexing, video browsing, video and speech retrieval, phonetic speech retrieval.

1. INTRODUCTION

Inspite of the wide access to information offered by the web, multimedia is not ubiquitous today due to the complicated task of searching and browsing the audio and video content. The proliferation of powerful search engines that efficiently search large text collections on the web has resulted in the casual user feeling comfortable with the current search-and-browse methods. However, when it comes to video and audio, there is no universal consensus on how users should pose a query, what kind of information should be presented in the list of matches, and how to browse through numerous retrieved videos quickly and efficiently.

Information retrieval of unstructured data, such as text documents (in contrast to structured data such as relational databases), has been studied for more than three decades. As a result, there is a broad consensus on document indexing, query presentation, as well as the ranking methods of retrieved results. Appropriate extensions for hypertext and web pages exist and exploit the extra information in anchor text, in the graph structure of the links, and so on. The main reason that this has been so successful is that text documents are built of words. Words are the basic elements used to construct a text document and are the most natural way for a user to make a query. Hence it is logical to carry out all the steps of the document-indexing process using word representation. Methods like tokenization, stop-word removal, stemming, part-of-sentence tagging, and term weighting have become standard techniques in information retrieval (e.g., see [1]). In contrast, multimedia content, such as digital audio and video, is represented by temporal signals, composed of pixels and audio samples. Semantic information that could be expressed in few words is implicitly expressed by millions of color values. Yet, in most cases, the user would still like to make a query in words, seeking for highly semantic information. This is known

as the *semantic gap* of multimedia information retrieval [2].

Extracting semantic information from image, audio, and video is a very complicated task. Even manual cataloging and annotation of multimedia is difficult as it is important. It requires considerable human labor but is limited to what is considered relevant by the annotator at indexing time, rather than the specific user needs at query time [3, 4]. Manual annotation of a single image, to be useful, must be done at three different levels: pre-iconography, iconography, and iconology, using special lexicons [5]. Still, this provides only a partial solution of the real needs of image and video retrieval. Despite limitations and cost, manual annotation is still considered one of the best available ways to index videos and images. It is used in commercial image and video collections (e.g., the Virage video logger [6]). This is significantly in contrast to text retrieval, where each of the four information retrieval phases: feature extraction, indexing, similarity matching, and ranking, are well defined and successfully automated. Invariably, each component of the multimedia data, image, text, video, and audio supports a different feature space and retrieval method specific to the media type. Therefore, information fusion is a critical aspect of effective multimedia retrieval.

Search in multimedia data requires content-based retrieval, where the data content is examined for the presence or absence of an object, event, related spoken words, and so on. Multimedia retrieval has been classified into content-based and semantic approaches [7]. Content-based techniques rely on an example or a physical low-level description of the information sought. For example, "a red object moving from the upper left to the lower right corner over a white background," would be expressed through sample images/videos or by using a special graphical user interface that include basic art tools [8, 9, 10, 11, 12]. Semantic approaches rely on the actual semantic content of the media [13, 14], "downhill skiing," for instance. The latter is more desirable from the user's point of view, but it is still uncertain how a machine could translate the query between these two very different forms or retrieve them in a similar manner.

2. THE COMBINED APPROACH: SEARCH THE SPEECH, BROWSE THE VIDEO

Understanding the similarities and differences between text search and video search would help design a multimedia search strategy that would satisfy the user's informational needs. With text search, a user typically poses a short text query (2–3 words) and expects to quickly retrieve one, or at most, a few pages of relevant results. Then the user briefly looks at the result set and either browses a few of the documents to find the one of interest or refines the query and repeats this process. There are three major tasks that take place during this process: query formulation, browsing the result set, and browsing individual documents within the result set. Each of these tasks is distinctly more complex for multimedia in comparison with text. Query formulation might include text and/or images, audio, video, and graphical inputs [14, 15]. While ranked results of text searches are presented by pragmatic short lists, there is a considerable variation in the information presented in a multimedia result list. Finally, after selecting an individual video to browse, the process of browsing that video is significantly more complex than visually scanning a text document.

The design approach for each major task was chosen in order to minimize the time required to satisfy the user's informational needs. Keyword search was selected for query formulation at a semantic level. Ranked results are presented using textual metadata that includes the video title, time offset, the matching spoken keywords, and the relevance score. A new media browser was designed to allow rapid browsing of relevant video segments.

Searching the speech transcripts using free text queries proves almost as efficient as searching text [16]. However, browsing through video is more time consuming than browsing of text, as the user has to play and listen to each of the retrieved videos one by one. With text, a quick glance at the results page is often sufficient to filter the information. With video, it is more efficient to browse the visual part. For example, consider a video collection of talks and presentations. A few storyboard pages, where each comprises ten or more key frames, can cover one hour of presentation, mainly showing the slides presented during the talk. To address the problem of browsing an individual video, a *synchronized viewer* is introduced. It begins playback at the time offset corresponding to the match and supports synchronous switching back and forth between various browsing modes, referred to as *views*.

The guiding theme of this work is "search the speech, browse the video." Video and audio are two parallel media streams that are tightly coupled by a common time line. We take advantage of the two parallel streams in a complimentary manner: we use the audio stream for searching and the video stream for quick visual browsing. Such functionality fulfills the user's informational needs expediently. This paper describes the system architecture, search algorithms, and synchronized viewer interface that supports the "Search the speech, browse the video" principle. The rest of this paper is organized as follows: Section 3 summarizes related work on multimedia information retrieval. Section 4 describes the overall system architecture. Sections 5 and 6 provide detailed description of the synchronized browsing and of the speech retrieval parts of the system, respectively. Section 7 presents a summary and discussion of this work in a broader context of multimedia retrieval.

3. RELATED WORK

Many researchers have been working on video indexing and retrieval techniques, using both audio and visual content. This has evolved as a multidisciplinary effort, which includes a wide range of research topics from computer vision, pattern recognition, video analysis and summarization, speech recognition, natural language understanding, and information retrieval (see, e.g., [7, 17, 18, 19, 20, 21, 22, 23]).

The early approach to video retrieval was to apply existing image retrieval techniques to key frames extracted from the video. The basic approach posted an image as a query, and retrieved similar images. This approach has two significant limitations. First, in most situations, the user does not have the image handy to formulate the query. Hence, it is more appropriate for browsing than for searching. Second, the approach does not take advantage of the temporal information in a video. Most of the work in content-based image retrieval (CBIR) has been done in the so-called *feature space*, composed of color histograms, color layout, color blobs, texture, and edge-based descriptors [15, 17, 18, 24, 25, 26]. While most of these methods use only global features, representing the entire frame, a few methods support also the segmentation and search of objects and faces within the frame [7, 10, 11, 26]. Even then, these methods still lack the ability to identify segments as objects and to associate them with any higher semantic meaning, such as object type, position in the scene, action, events, and so on.

Semantic-level indexing was first demonstrated in specific video domains. Examples include statistical methods trained for domain-specific detection and classification of events. See [27] on classifying tennis strokes to six categories, [28] on event detection in soccer, [22, 29] on human motion classification, like walk and hand gestures, and [12] on motion trajectories. Other works use object recognition for indexing, such as Qian et al. [30] who uses multiple modalities for indexing and retrieval.

Naphade et al. [31] introduced multijects as a generic representation that binds high-level concepts with low-level features using a Gaussian mixture model. Different multijects are used in a probabilistic framework that captures correlations between them. In the reported work, no object segmentation was used. Lacking segmentation, the low-level features are obtained over the entire image. The features represent the whole visual scene and make it difficult to distinguish between different objects or between objects and the background.

In [32], the authors review spatial segmentation of video objects. These regions can be used for object indexing. Having a user in the loop can also be used for labeling these objects. For example, such semantic information can be inferred from the user's feedback while searching and browsing images for a specific topic [33]. Semantic information is also extracted in [9] using motion analysis and object segmentation. However, nearly any mapping from low-level feature space into words and concepts is based on an implicit continuity assumption: that each concept forms nice clusters of images in, say, a color-texture space. This generic approach has been successfully demonstrated on a few classes of image and video, such as airplanes, red roses, tigers, and explosions. Yet, this approach has to be proven in the much wider context of semantic concepts.

Searching the audio transcript of the video using the familiar metaphor of free text search has been studied in several projects [19, 23, 34, 35]. In this case, automatic speech recognition (ASR) is applied to the audio track, and a time-aligned transcript is generated.[1] The indexed transcript provides direct access to semantic information in the video. Phonetic speech retrieval is also used for speech retrieval, usually in addition to the ASR. A *phoneme* is defined as any of the abstract units of the phonetic system of a language that correspond to a set of similar speech sounds which are perceived to be a single, distinctive sound in the language. Whereas phonemes are defined by human perception of sounds, the subword units used by the ASR systems are generally data derived and are commonly referred to as *phones*.

The Informedia research project [23, 34, 36] has created a terabyte digital library where automatically derived descriptors for video are used to index, segment, and access the library contents. Informedia combines speech recognition, image processing, and natural language-understanding techniques. Combined word and phonetic retrieval was also explored, where an inverted index for a phonetic transcript comprised of phonetic substrings of three to six phones in length was used. During retrieval, the word document index and phonetic transcription index were searched in parallel and the results merged.

Cambridge University, in collaboration with Olivetti Research Laboratory (ORL), has developed the Medusa networked multimedia system. It is in use on a high-speed ATM network for a video mail application based on word-spotting using a 35-word indexing vocabulary chosen a priori for the specific domain. They developed retrieval methods based on spotting keywords in the audio soundtrack by integrating speech recognition methods and information retrieval technology to yield a practical audio and video retrieval system [37].

The National Institute of Standards and Technology (NIST) conducts and sponsors the annual Text REtrieval Conference (TREC). This framework promotes information retrieval research for realistic applications by providing large training and test collections, uniform scoring procedures, and thorough evaluation of submitted results [38]. The spoken document retrieval (SDR) track, which was active from 1997 until 2000, explored content-based retrieval of excerpts from archives of speech recordings using a combination of ASR and information retrieval technologies [16]. The SDR track was followed by the current video track, started in 2001, with emphasis on CBIR [14]. In 2002, the second video track promoted a new approach to video search. Ten specific concept detectors were applied to the videos, including detection of faces, people, indoor/outdoor, text within image, and so on. Their results were combined to retrieve high-level semantic topics. Each of those detectors was the result of dedicated efforts in the domain of computer vision research.

Video browsing seems to have received less attention than video retrieval. However, the browsing efficiency problem arises whenever a system is built over a large collection of videos. The Informedia project offers three ways to view search results on video: poster frames, filmstrips, and skims

[1] In cases where closed captions are available, they are often being used instead of or in addition to the ASR.

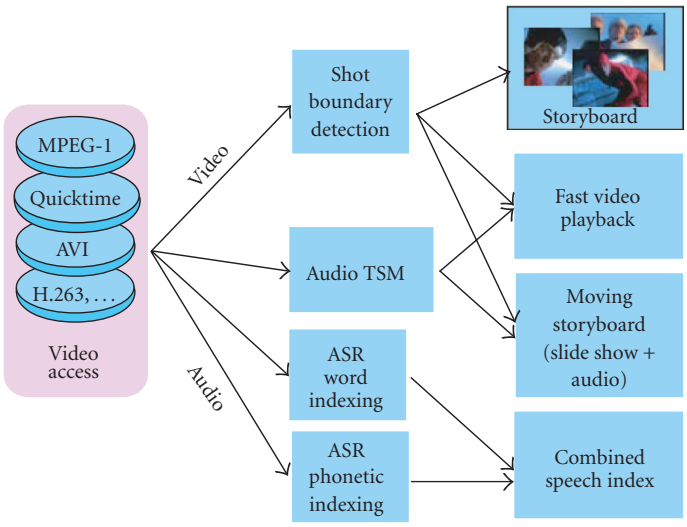


Figure 1: CueVideo indexing architecture.

(video summaries). The poster frame view presents search results in a poster frame format with each frame representing a video paragraph. The filmstrip view reduces the need to view each video paragraph in its entirety by providing storyboard for quick viewing [34]. The most relevant subsections of the video paragraph are displayed as key scenes, and key words are clearly marked. The Collaborative and Multimedia Group at Microsoft Research made considerable work in multimedia browsing, studying both the technical aspects and the user behavior [39, 40, 41]. This includes video summarization, audio speedup, collaborative annotation, and the usage patterns of such technologies.

Chang et al. [9] developed the WebClip system for the search and browse of video for education and tested it in several K-12 schools. Their main focus was visual indexing and retrieval. The system includes most of the various components described above and includes content-based, semantic image retrieval, object segmentation, motion trajectories, textual annotation, and advanced browsing capabilities such as a client video editor. Indeed, they report on the difficulties of automatically extracting semantic knowledge from images, and state, with regard to content-based indexing, that videos have an advantage over images. The reported client-server implementation requires high communication bandwidth over a local network.

4. SYSTEM ARCHITECTURE

A prototype system to support the "search the speech, browse the video" paradigm was developed over five years in the CueVideo project at the IBM Almaden Research Center. The architecture consists of two parts: an off-line indexing phase, and an on-line retrieval system that supports real-time search and browse. The indexing phase, shown in Figure 1, begins with demultiplexing the original video into video and audio streams. The video component is directed to a shot boundary detection module [42], which detects scene changes, and for each shot generates a representative key-frame image. The audio component is directed to three modules. The first speeds up the audio and generates multiple audio tracks at preselected speeds [40, 43]. The audio time scale modification (TSM) algorithm [44] preserves the original speech pitch and intonation. The second module processes speech and generates a word index. It is based on ASR, followed by word tagging, stemming, and indexing. The third module generates an ASR-based phonetic index. As a result, several searchable speech indexes are created, including an inverted word index, a phonetic index, and a phrase glossary index. In parallel, several browsable views are generated, including a storyboard, slideshows with audio at different speedup ratios, animation, and a nonlinear fast playback [42].

The CueVideo architecture for real-time client-server retrieval system is shown in Figure 2. The backend is composed of three servers: web, media streaming, and speech retrieval. In the scheme of Figure 2, the speech retrieval server is a part of the application server. Implementation of the speech retrieval server provides a TCP interface and an API interface, allowing two-tier or three-tier web applications to be implemented [45]. A prototype system was built and deployed on the corporate intranet, using three-tier configuration in which the speech retrieval server runs on one computer and can simultaneously serve multiple web servers that run on different machines. The speech retrieval server can be distributed over a cluster of computers to support a high volume of video data and a large number of simultaneous queries. The media streaming server is independent of the web server and can be hosted on the same computer or on a separate one.

The client user interface runs on a standard web browser and can work with different media streaming plug-ins. It supports free text queries. The queries may contain any keywords, in-vocabulary (IV) and out-of-vocabulary (OOV)

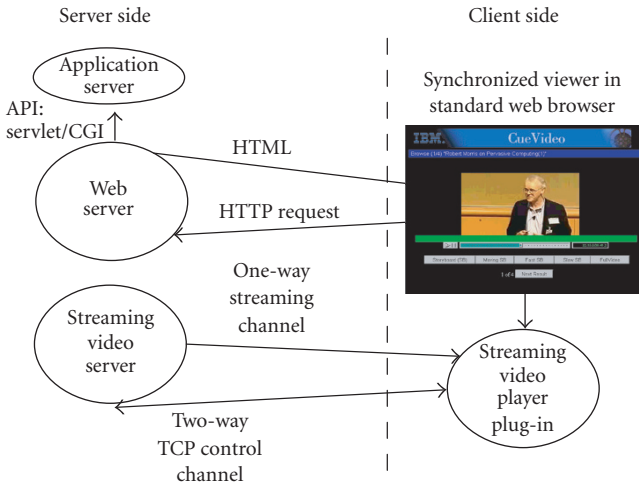


FIGURE 2: CueVideo media retrieval and browsing architecture, implemented in a standard Internet server-client environment.

words, phrases, pronouns, and acronyms to search the speech index of the video collection. The query is submitted as an HTTP request to the web server. A CGI program is invoked and the request is processed, resulting in a speech query being submitted to the speech retrieval server, which performs the speech search and returns a ranked list of results. The results are rendered in two forms. The first is a text table of the matches. Each match is listed with the video name, the begin- and end-times of the matched segment and corresponding words found in the speech. The second rendering form is the video browser with multiple synchronized views.

5. VIDEO BROWSING USING MULTIPLE SYNCHRONIZED VIEWS

Browsing video documents, unlike browsing text documents, requires an additional level of point and browse within the document. Loading the entire video document to the client is impractical and will not help the user find relevant points within the video. However, most standard streaming video players can start video playback at any given time offset. When browsing search results, the video starts to play at the offset that corresponds to the first match. The user might want to view this segment, browse other parts of the same video, or skip to the next match, either within the same video or in a different video.

Streaming media players provide a time bar, which can be used to seek any point in the video. However, this bar does not provide any information about the video content. Thus browsing becomes a tedious and inefficient sampling of different points in the video. A browser that supports multiple, synchronized views was designed to address this problem. The content of the video can be presented in different visual forms, such as

(i) *storyboard*: a table of thumbnails of key frames presented in chronological order and labeled with the SMPTE shot times;

(ii) *moving storyboard (MSB)*: a slide show composed of representative key frames, fully synchronized with the original audio track. Each key frame is shown for the entire duration of the associated shot. It is extremely useful for browsing videos of presentations and talks over low bit-rate connections, where static key frames capture the speaker's slides at a much higher quality than that of a streaming video;

(iii) *fast/slow moving storyboard (fast MSB)*: a slide show with fast audio. The audio is sped up using pitch-preserving audio speedup algorithm [44]. Similarly, a slow MSB is also generated;

(iv) *animation*: a very quick slide show without audio, where each of the key frames is shown for a fixed short period (e.g., 0.6 seconds). This is a very fast, nonlinear time compression of the video, giving the same duration to long and short shots;

(v) *accelerated fast playback*: very fast video playback without audio. The speedup ratio depends on the video content. At the beginning of a shot, the speedup ratio is 2.0, and after 0.5 seconds, it starts to gradually accelerate to twenty times faster or more. It resets back to 2.0 at the beginning of the next shot. This ensures that short shots are not missed. The ramp-up period gives the user enough time to fixate on some of the shot content, after which the user is able to follow the very fast playback of the rest of the shot, which is perceived in very fast motion.

These different representations are called *views*. Some views, such as storyboard and animation, do not have audio. Visual-only views are generally much faster to browse. However, in some cases, the information is mainly found in speech and audio. In such cases, a view with audio must be used, or text from speech can be added to visual views. Views with audio speedup allow viewing relevant video and audio segments in a shorter time. A user study on comprehension of sped-up audio [43] found that (untrained) users prefer to listen to an audio 10–20% faster than at the original speed. On average, users feel comfortable capturing content at 50% faster playback and start to miss content when played back is faster than 70%. These rates depend on the video content, vary between individuals, and can be dramatically increased by practice. It is interesting to note that in a comparison between the comprehension of full video, slide show, and audio only, some users show a significant advantage to full video while others show a significant advantage to audio only. That is, the visual component does not always help in understanding the speech. This is another reason to offer the end-user flexibility in selecting the view best suited for the type of video content. Verbal feedback received from the users of this system [46] indicates that the most popular views are storyboard for rapid browsing and fast MSB with audio speedup. Other user studies by Omoigui et al. and Li et al. [40, 41] report quantitative evaluation of video browsing and usage patterns of audio speedup and compare alternative implementations.

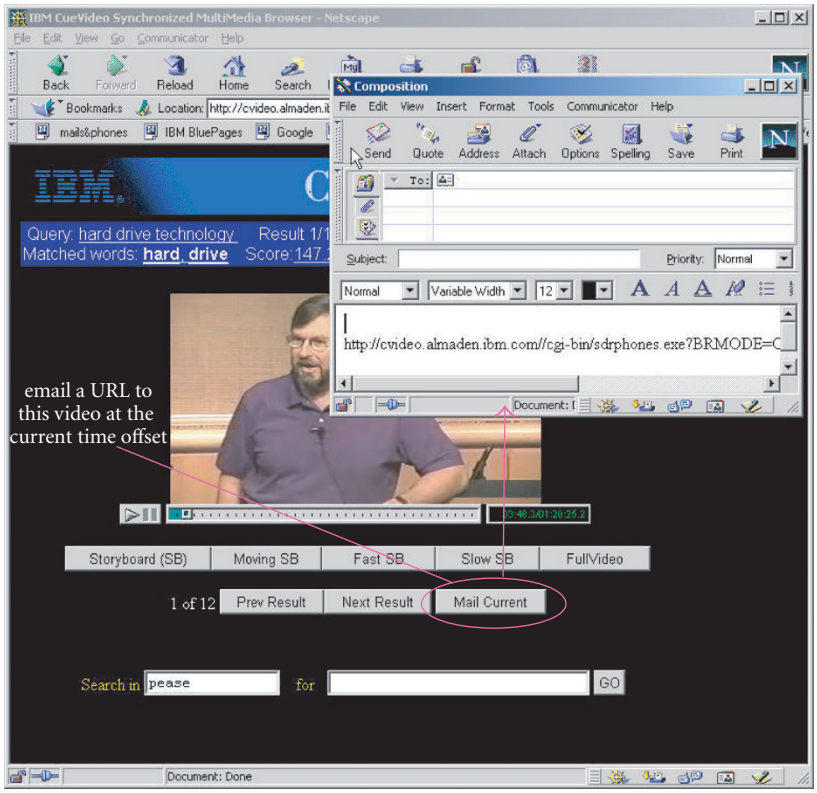


Figure 3: Video browser with multiple synchronized views, controlled by the extra buttons below the player. The text query and the result-matching words, video name, offset and score are displayed at the top.

The user is able to choose which view is most relevant to the task. Moreover, the user is able to switch from any view to any other view at any point in time while browsing. The new view starts to play from the point that corresponds to where the user left the previous view. This provides context continuity in browsing, regardless of the switch between views. This concept is referred to as browsing with multiple synchronized views, or *synchronized multiview playback*. This becomes extremely important for complex information-seeking tasks in long video clips where a combination of different browse modes may be used to filter and find the desired information rapidly. It allows, for example, quick switching from visual browsing, like animation, to a slide show with fast audio or to full video.

Figure 3 shows an example of a query result as it is browsed in the synchronized multiviews browser. The browser page consists of an embedded RealMedia player, and additional buttons to playback each of the browser views: storyboard, animation, different MSBs and full video. In addition, the browser page contains "Next Result" and "Previous Result" buttons to allow quick navigation thru search results to the next relevant match whether it be within the same video clip or in a different one. The user can easily switch back and forth between the different browse modes, starting the playback in a new browse mode from the point in time where a previous browse mode was left off. The "Mail Current" button invokes a mailer window with a new note containing a link to the currently playing video at the current time offset. This is immense, for example, in video applications for remote education. A student can refer in his new mail to a certain point in a lecture and ask questions about it.

There are several engineering challenges in implementing a synchronized browser [45]. The main difficulty is caused by the fact that the application server has no direct communication with the streaming media server. All communications must be performed through the client (Figure 2). A standard multimedia streaming player plug-in establishes two connections via communication ports with its media streaming server. The first is a two-way TCP connection to send commands from the player to the server such as Play, Pause, Stop, and so on. This channel also supports authentication and logging functionality on the server. The second connection is a one-way downstream data channel established from the streaming media server to the player plug-in to stream the media. In order to switch from one view to another in a synchronized manner, the browser must first determine the current point in the current view and compute the corresponding point in the new view. Then it switches to the new view, seeks the new point, and begins to play. This computation of the corresponding point requires video-specific information about the current view and the new view. In order to prevent extra delays in switching between views and matches, all relevant information and switching logic is coded in JavaScript and embedded into the DHTML

browser page. This includes the query results and the timing information that is required to allow synchronized switching between views. Hence switching between views is performed between the client and the streaming media server without issuing any additional HTTP requests to the web server. More details can be found in [45].

6. SPEECH INDEXING AND RETRIEVAL

The speech retrieval system includes a word-based index, a phrase index, and a phonetic index. A search is performed in all indexes relevant to the query and results are merged. For example, an OOV word is only searched in the phonetic index, while an IV word can be searched in all three indexes.

6.1. Word-based indexing of an ASR text

The indexing is performed in several steps. First, a speech recognition system is used to transcribe the audio and to generate a continuous stream of timed words. The IBM real-time engine is used with the broadcast news vocabulary of 60,000 words [47]. The word error rate for a large vocabulary speech-recognition task on prepared speech (as opposed to spontaneous speech) from anchors in a studio is reported to be around 19%. In general, for a wide variety of real-world speech data that includes combinations of speech with background noise, degraded acoustics, and nonnative speakers in a real-time speech-recognition system, error rates can vary between 35% and 65% [34, 48, 49, 50]. The rest of the ASR-based speech indexing process includes stemming, stopwords removal, morphing, and POS tagging (For more information about these standard techniques, see, e.g., [1]).

The retrieval system first loads the inverted index and the precomputed weights of each of the nonstop words. Given a query q, a single-pass approach [51] is used to compute the relevancy score for each document. The relevancy score $S(d, q)$ is the score of document d with respect to query q and is given by the Okapi formula

$$S(d,q) = \sum \frac{(C_q(q_k) * C_d(q_k) * \text{idf}\,(q_k))}{a_1 + (a_2 * (l_d/l_{\text{bar}})) + C_d(q_k)}, \qquad (1)$$

where k is the summation index over the query terms, q_k is the kth term in the query, and $C_q(q_k)$ and $C_d(q_k)$ are the counts of the kth term in the query (usually one) and in the document, respectively. The document length and the average document length are denoted by l_d and l_{bar}, respectively. The Okapi constants are set to $a_1 = 0.5$ and $a_2 = 1.5$. The inverse document frequency $\text{idf}(q_k)$ for the query term q_k is computed by

$$\text{idf}\,(q_k) = \log\left(\frac{(N - n(q_k) + 0.5)}{(n(q_k) + 0.5)}\right), \qquad (2)$$

where N is the total number of documents and $n(q_k)$ is the number of documents that contain the term q_k.

Each word in the query string is tokenized, tagged, morphed, and then scored using the Okapi formula above. The scoring function takes into account the number of times each query term occurs in the document $C_d(q_k)$, normalized with respect to the length of the document l_d. This normalization removes bias that generally favors longer documents since longer documents are more likely to have a larger number of instances of a particular word. The $\text{idf}(q_k)$ term favors rare words over common words. Those rare words are more discriminative during the search.

We refer to this retrieval method as the *baseline* system. The baseline system can be extended in several ways, one of which uses query expansion and the other uses phonetic retrieval.

6.1.1 Query expansion

Query expansion is the process of adding words and terms to a query in order to extend the scope of the search to obtain a better match between the query and document representations. Speech recognition transcripts include insertion, deletion, and substitution errors. We evaluated three different query expansion methods intended to offset the dominant deletion error [50]. In the first method, we cluster fixed-size unit segments from a given speech transcript. Then segment cluster labels are generated to extract potentially important multiword concepts and names that may convey the underlying topic structure in the transcript. Where single-word labels are not meaningful out of context (e.g., large, system, index, finger, etc.), multiword noun phrases often convey the underlying information (e.g., system architecture, index finger, etc.). We use these multiword cluster labels to expand the original query [52, 53]. For the second method, we use the multiple recognition hypothesis generated by the speech recognizer to enhance the query with additional query terms. The third method uses a thesaurus. This technique is well known and has yielded mixed results in the past. Table 1 shows an example of an expanded query using each of the methods described above for the query *audio selector panel*. We evaluate retrieval using our query expansion methods against the baseline retrieval system. The test data used was a twelve-hour corpus of corporate communications/distance learning data consisting of ∼300 segments. Approximately fifty test queries were used to calculate precision and recall. The relevant segments per each query (or ground truth) were manually computed by observing all the videos.

Table 2 lists the average recall within a single video, the average precision for each method across the video collection, and the average precision within a single video. To summarize, the cluster-labels query expansion resulted in modest improvements of 3% in recall and 5% in precision for search within a single video. However, none of the methods we experimented yielded any significant improvement for retrieval *across* the video collection.

6.2. Phonetic speech retrieval

Phonetic speech retrieval addresses retrieval of OOV words, acronyms, phrases, and expressions, as well as overcoming some of the speech recognition errors for IV words. Several other researchers have explored the use of subword

Table 1: Sample expanded queries for query "audio selector panel."

Thesaurus	Cluster labels	N-Best
audio, sound selector, switch, panel	audio selector panel, audience elector panel, audio accessory unit, pilots call panel	audio adios, panel tamils pavel paddle petals, kennel kemp

Table 2: Query results using four query expansion methods compared to baseline system.

Spoken document retrieval system—query expansion method	Recall	Avg. precision across video collection	Avg. precision within single video
Baseline System (no query expansion)	0.74	0.93	0.7
Thesaurus	0.72	0.93	0.7
Cluster Labels	0.77	0.93	0.75
N-Best	0.65	0.93	0.63
Cluster Labeling and N-best	0.71	0.8	0.66

representations based on phonemes as index terms with varying degrees of success [19, 54, 55, 56]. The accuracy of phone recognition is limited, particularly in the case of short words [54, 55]. However, for the purpose of retrieval of OOV words and in instances where the confidence level associated with the recognized words is low, there is considerable benefit in combining phonetic information with word-level information. The use of phonetic retrieval also makes "sound-like" retrieval applications possible.

During indexing, a phonetic transcription of the input audio is generated based on the ASR timed speech transcript [35]. The equivalent phonetic sequences are automatically generated using the US English phone set. We refer to this as *phonetic transcript* since it is convenient to be considered as a string of phonetic characters, taken from an alphabet of 51 phones, namely, the standard American English phonetic alphabet. The phonetic transcript is a compact representation of the sounds of spoken words.

During retrieval, the text query is first converted to a string of phones (see, e.g., Lawrence's metaphones algorithm [57], not to be confused with our notion of metaphones). Then sound-like matches are retrieved in the phonetic space. Hence retrieval is based on string-matching techniques for partial string matching in the presence of high phonetic error rate. This introduces a known trade-off between accuracy and efficiency. Two main components were developed to address this trade-off: an efficient indexing of erroneous strings and a sound-like phonetic string-matching algorithm.

6.2.1 Modeling sound-like similarity of phones: the confusion matrix

For a given phonetic alphabet, we define the *phone confusion matrix* to model the probability of a phone to be mistakenly recognized by a phone recognition system as a different phone. For the US English phone set of 51 phones, we define the confusion matrix C. Each element in the matrix C_{ij} represents the probability of missrecognizing phone q_j as phone q_i, that is, $C_{ij} = P(q_i|q_j)$. Additional rows and columns are used to model insertion, deletion, and pauses errors. Higher-order statistics of confusion can be also used such as triphone confusion probabilities.

6.2.2 Using metaphones for indexing

The phonetic transcript is indexed by using phone triplets as keys (denoted as *three phones*). Note that without special care, a single-confused phone in the phonetic transcript would prevent the system from finding any of the three keys that contain it. Due to the high phonetic error rate, system recall would be affected severely. Based on the content of the confusion matrix, seven groups of phones were identified, denoted as *metaphone* groups. A metaphone group consists of phones that are more likely to be confused with each other. Each group contains between two and ten similar phones, as shown in Table 3. For example, the phones B, BD, DD, and GD form one metaphone group. At indexing time, each key that contains one or more phones from metaphone groups is indexed using its metaphone representation, where the original phones are replaced with their metaphones. All records located by a given key are stored in a linked list, pointed from a table where each entry of the table corresponds to a three-phone key. This indexing method is rather efficient.

6.2.3 Measuring sound-like similarity: the Bayesian phonetic edit distance

Given a query term $q = q_1 \cdots q_n$ and an observed phone sequence in the transcript $o = o_1 \cdots o_m$, the edit distance is used to define the metric of sound-like similarity between the two phonetic strings. First, Bayes rule is used to compute the probability of an observed phone o_i origin from a (possibly confused) phone q_i,

$$P(q_i|o_j) = P(q_i)P(o_j|q_i)/P(o_j), \tag{3}$$

Table 3: Metaphone groups.

Metaphone	Metaphone group
Atl	AA, AE, AH, AO, AW, AX, AXR, AY, EH, ER
Ctl	CH, JH, SH
Etl	EI, IH, IX, IY
Gtl	B, BD, DD, GD
Th	TH, F
N	N, NG
G	G, D

where $P(o_j|q_i) = C_{o_j,q_i}$, $P(o_j)$ is derived from statistics over a large corpus of data and $P(q)$ drops once we normalize the score. When comparing two sequences, a one-to-one phonetic correspondence between two phonetic strings is unlikely to be found. Deletions and insertions are common errors in phonetic transcripts. Therefore, the string similarity is modeled using edit distance, more specifically the end-space free, weighted edit distance (for string matching overview, see [58]). An edit is a sequence of phone operations that consist of substitution, insertion, and deletion. The cost of editing o and converting it to q is the sum of the costs, or weights, of the required phone operations. Those are defined by the log probabilities as modeled by the confusion matrix. Hence, this cost resembles the likelihood of the editing sequence. It is the joint likelihood of its editing steps, assuming those are independent random variables.[2] The edit distance is the maximum likelihood among all possible edit sequences that convert o to q. It can be efficiently computed using dynamic programming [58]. Let $E[i][j]$ denote the cost of editing the prefix $q_1 \cdots q_i$ to the prefix of the observed sequence $o_1 \cdots o_j$. The values of $E[i][j]$ can be computed using dynamic programming with the updating formula

$$\begin{aligned} E[i][j] = \min \{ & E[i-1, j-1] + \log(C_{o_j q_i}), \\ & E[i, j-1] + \log(\mathrm{del}(o_j)), \\ & E[i-1, j] + \log(\mathrm{ins}(q_i))\}. \end{aligned} \tag{4}$$

Special attention must be paid to end-space free and partial substring-matching criteria. Note that at this point the result is not a metric anymore, because even the substitution of a phone with itself is scored with a nonzero log likelihood. To derive the final score of the match, the result is normalized by q's self edit distance. Hence the part of the equation that depends on the query $P(q)$ is eliminated from the score. This represents the log-likelihood ratio between the best edit found and the hypothetical optimal match of the query to itself. Hence the normalized score of an exact match is 1.0, and any other match would score between 0.0 and 1.0, with a higher score for a better match.

[2] In the current work, we study other models, like triphones and MRF, which also capture the dependencies between phones.

6.2.4 Phonetic retrieval of a text query

The phonetic word spotting consists of two stages. Given a query word, the phonetic representation of which is generated. Multiple triphone keys, each composed of three consecutive query phones, are extracted. The lists of (document, offset) candidate records found in the phonetic index for each of the keys are merged into a single list of candidates. Then, each candidate is compared with the phonetic query and is ranked, based on its Bayesian phonetic edit distance.

The final step is to combine the text retrieval candidates with the phonetic candidates. We could suggest different ways of combing these two ranked lists using various information fusion techniques. A very simple method was implemented, which gives higher preference to word-based matches over phonetic matches. This eliminates the possibility of phonetic retrieval to add a lot of false positives to the combined list, which would affect the retrieval performance. This conservative approach guarantees that speech-recognition word-based retrieval performance for IV words would not be hurt, a concern often mentioned in this context. As found in the experiments, the approach helps improve overall IV results. For OOV words, the phonetic candidates are the only candidates. As such, they are naturally listed by their phonetic scores.

The ranked list of candidates is truncated at a certain score threshold. While this is not required for a word-based index, it is inevitable for phonetic retrieval. The list of candidates include many false positives which happen to have three or more consecutive phones in common with the query. An adaptive threshold is computed from a histogram of the scores trying to maximize recall without adding too many false positives. This mainly depends on the top scores which indicates whether an exact match or nearly exact match was found. A fixed minimal threshold is applied to all other cases.

6.3. Experimental evaluation of speech retrieval

Experimental results for SDR (i.e., not word spotting) were reported in detail over the last several years in the SDR track of the TREC series of conferences [16, 48, 59, 60, 61]. A common theme from all previous SDR experimental work is that *multiword queries* were used in the test query set, *whole-story* segments were retrieved, and *relevance judgments* were made by humans in order to identify the relevant documents for each query. The task and evaluation criteria of this work are different from those mentioned above. The query set consists of single-word queries and the retrieved set should contain all occurrences of a query word in all video documents. The evaluation measure is based on the word's time-of-occurrence, using an objective ground truth which consists of a complete manual, time-aligned, accurate transcription of the speech. A match is considered correct if and only if the exact word was said within a two-second window of the retrieved time-of-occurrence. This time window tolerates imprecise word-times in the ground truth, which only provides times at sentence-level granularity. An inexact word match or a larger time difference results in a false positive and/or a miss.

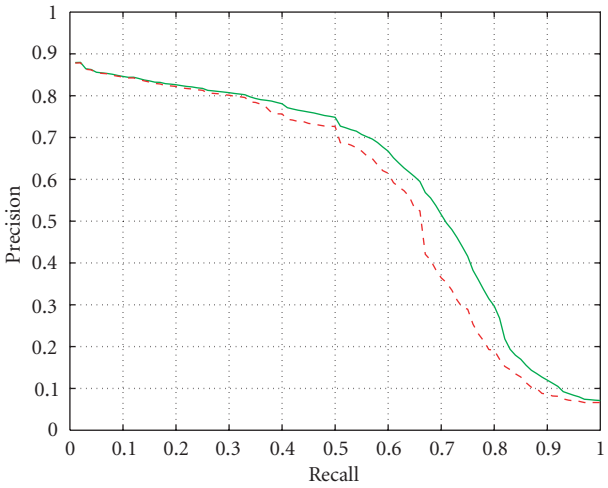
The test collection is based on 100 hours (1.04 million spoken words) of HUB4 data [62] where ASR word error rate is about 35%, of which 3% is due to OOV words. This data, accompanied with ground truth timed manual transcript, is traditionally used to assess speech recognition quality. It was found most suitable for the WS task, more than standard SDR data. The speech contains 24,018 different words, of which 17,955 are IV words and 6,063 are OOV words (after stop-words removal). Although 25% of the different words are OOV, they only occur about 3% of the time. Still, these words are of special significance for speech retrieval tasks as they are very distinctive and include many names of people, places, products, companies, and so on. The exhaustive queries test set consists of all the words that are listed in the ground truth transcription, namely, 24,018 different queries. These queries are divided into IV/OOV groups and are further subdivided by the number of phones they contain. The number of phones is an important factor (see, e.g., [63]). In general, the longer the phonetic query is, the more accurate is the result. This is evident in Figure 5.

Each of the queries was processed independently. The results were combined using the average precision at recall points method. The graphs in Figure 4 present the precision as function of recall for IV words, nine and twelve phones long. The recall is normalized with the number of ground truth word occurrences, such that a recall of 0.6 means the recall of 60% of all occurrences of the query words. This figure shows an improvement of 5–12% in the precision of combined phonetic retrieval (solid line) compared to speech recognition alone (dashed line). The largest improvement, between 10–15%, is achieved for word lengths in the midrange, about nine phones long. The precision of both baseline and phonetic systems increase for longer words as expected from the error model in [63].

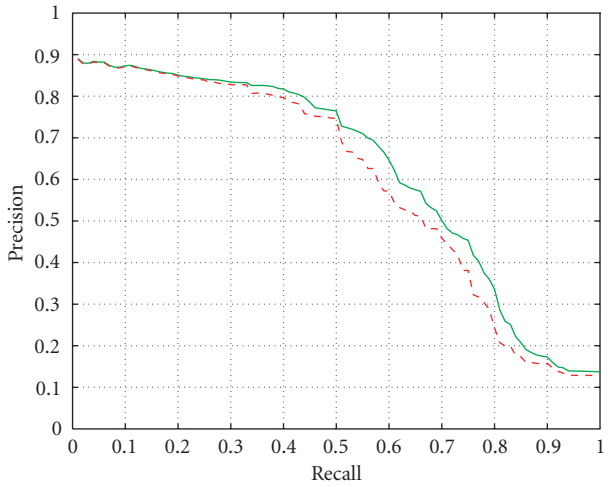
The results for OOV words were not as good. Average precision is 20–25% at 50% recall. Note that most of the OOV words in this particular data set are mentioned only once. Thus the task of finding this single occurrence within 100 hours of audio is not an easy one. A large part of the retrieval error is due to the high rate of phonetic recognition errors. Another part corresponds to indexing misses. The last part is due to the limited ability of the score function to accurately model sound similarity. The number of false positives increases linearly with the site of data and has a larger impact on the precision of rare words.

The overall performance is summarized in Figure 6. The graph presents the average precision recall for all 24,000 queries over the entire data. A significant 15% improvement in precision is gained at 70% recall.

There is a trade-off between the number of indexing errors and the amount of time spent on query processing. The larger the number of candidates considered, the higher the recall. However, this requires longer query processing time. The results reported above are obtained at an average processing time of 0.5 seconds per query for retrieval in 100 hours of video. Performance tests on data sets of 10, 100, and 500 hours are reported in Table 4. Special attention was paid to developing appropriate data structures. Minimizing the number of page faults while reading from the disk was required to achieve these fast-processing times. More information can be found in [64].



(a) Average precision-at-recall points for nine-phone-long IV words (1981 words, 18739 occurrences). Combined phonetic retrieval (solid line) improves up to 12% over ASR retrieval alone (dashed line).



(b) Average precision-at-recall points for twelve-phone-long IV words (482 words, 2706 occurrences). Combined phonetic retrieval (solid line) improves up to 5% over ASR retrieval alone (dashed line).

Figure 4

7. SUMMARY AND CONCLUSIONS

Video retrieval and browsing is a complicated task. What makes it so much more complicated than text retrieval is the implicit, hard-to-penetrate information represented in audio and video signals. Further, audio browsing is much less efficient than visual browsing of text, images, and other visual media. Special attention must be given to efficient browsing of search results and efficient intradocument browsing of

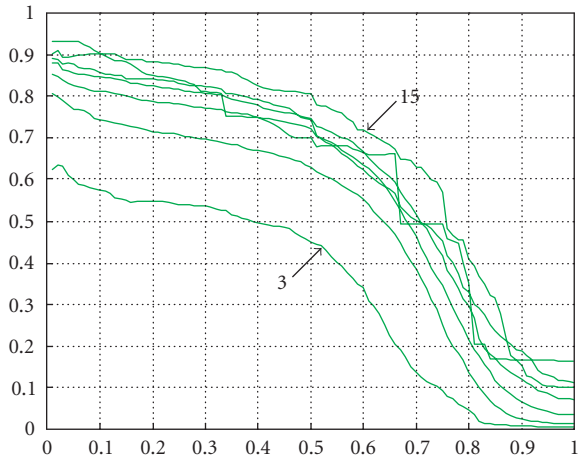


Figure 5: Average precision-at-recall points of the combined ASR and phonetic speech retrieval for IV words, 3, 5, 7, 9, 11, 13, and 15 phones long. Shorter words have lower precision.

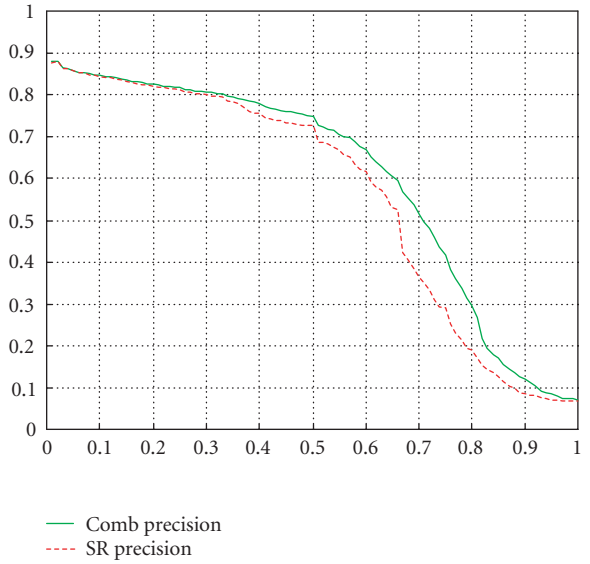


Figure 6: Average precision-at-recall points for all 24,000 words. Combined phonetic retrieval (solid line) improves up to 15% over ASR retrieval alone (dashed line).

retrieved video matches. Most content-based video retrieval methods are appropriate for browse purposes but lack the ability, as yet, to search for any kind of semantic information. On the other hand, most of the SDR methods are appropriate for semantic searches through automatically transcribed speech but provide no efficient means for browsing.

This paper proposes a combined approach: *Search the speech, browse the video*. This approach takes advantage of search capabilities from SDR and combines them with efficient visual browsing. Combined phonetic and ASR-based speech retrieval allows searching of words, phrases, keywords, acronyms, and OOV words. Extensive experimentation results with 24,000 queries on 100 hours of video show

Table 4: Average query-processing times for different corpus sizes. (600 MHz P-III, 1 GB RAM)

Data set size	Average query-processing time
10 hours	0.18 sec/query
100 hours	0.41 sec/query
500 hours	1.94 sec/query

5–15% improvement in precision compared to retrieval using ASR alone.

A synchronized multiview browser was developed to support efficient video browsing. It allows the user to efficiently browse retrieved videos by quickly playing relevant video segments and switching between different views such as storyboards, slide shows, animation, and fast speech playback. The different views are synchronized and maintain the context and relative point in the video at the time of switching.

The system was used to capture three local conferences in video, index them, and create on-line video proceedings of the conferences. This collection contains 34 hours of video and was made available on the IBM Intranet. Verbal feedback from users of the system indicates that the most useful browsing features of the system were the offset playback from the storyboard for rapid browsing and the fast MSB with fast speech for listening. The speech search was found more useful for people who were familiar with the collection. Otherwise, users did not know what they could search for. However, after glancing at the conference program, they came up with appropriate search terms. A much larger and heterogeneous video collection, along with minimal user experience, could overcome this initial user reaction.

Although work in content-based image and video retrieval has made major progress in recent years, we remind ourselves of the broader context in which future systems might be exploited by their users—very much like text search is being used today. Examples of challenging topics and queries can be found in the search task of the NIST TREC first video track, held in 2001. A search topic such as "a lunar vehicle traveling on the moon" [14] provides a compelling example. Moreover, the broad range of semantic events and topics detection, as studied by Enser [3, 4] and others, is still in its early stages. One of many examples for such topics, posed by companies to commercial media agencies, was to find "a nice picture of a man and a woman, exercising, smiling, and not sweating." There is more work to be done before such queries can be handled by any automated indexing and retrieval system.

In 1999, at the Fourth Search-Engines meeting, Boston, there was still a debate between the manual cataloging and browsing fans and the searching advocates. In a sense, both won. Users do need both search and browse capabilities, and a combination of the two is the most powerful. Users are accustomed to browsing search results and taxonomies and then narrowing their search to "search only within this category." Sometimes a user searches for a company homepage and then browses its directory. Search and browse are two

seamlessly coupled operations for text. Yet, they become very distinct when we try to apply them to video and multimedia. Good annotation tools, extraction of semantic visual features, a common description scheme (like MPEG-7), and multimodal search tools are essential building blocks. However, we must pay close attention to user expectations, to the ways users adapt and systems change through the years, especially as history reveals in the text retrieval domain. Making multimedia search and browse work together is an important aspect in this area.

ACKNOWLEDGMENT

We would like to thank Heather Poyhonen for her great help with the manuscript.

REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, Reading, Mass, USA, 1999.

[2] X. S. Zhou and T. S. Huang, "Unifying keywords and visual contents in image retrieval," *IEEE Multimedia*, vol. 9, no. 2, pp. 23–33, 2002.

[3] P. G. B. Enser, "Query analysis in a visual information retrieval context," *Journal of Document and Text Management*, vol. 1, no. 1, pp. 25–52, 1993.

[4] L. H. Armitage and P. G. B. Enser, "Analysis of user need in image archives," *Journal of Information Science*, vol. 23, no. 4, pp. 287–299, 1997.

[5] S. Shatford, "Analyzing the subject of a picture: A theoretical approach," *Cataloging and Classification Quarterly*, vol. 6, no. 3, pp. 39–62, 1986.

[6] A. Hamrapur, A. Gupta, B. Horowitz, et al., "Virage video engine," in *Storage and REtrieval for Image and Video Databases V*, vol. 3656 of *Proceedings of SPIE*, pp. 188–197, San Jose, Calif, USA, February 1997.

[7] A. Gupta and R. Jain, "Visual information retrieval," *Communications of the ACM*, vol. 40, no. 5, pp. 70–79, 1997.

[8] S.-F. Chang, W. Chen, and H. Sundaram, "VideoQ: a fully automated video retrieval system using motion sketches," in *Proc. 4th IEEE Workshop on Applications of Computer Vision*, pp. 270–271, Princeton, NJ, USA, October 1998.

[9] S.-F. Chang, A. Eleftheriadis, and R. McClintock, "Next generation content representation, creation and searching for new media applications in education," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 884–904, May 1998, Special Issue on Multimedia Signal Processing, Part One.

[10] J. R. Smith and S.-F. Chang, "Querying by color regions using the visualseek content-based visual query system," in *Intelligent Multimedia Information Retrieval. IJCAI*, pp. 159–173, MIT press/AAAI press, 1997.

[11] Deng Yining and B. S. Manjunath, "NeTra-V: toward an object-based video representation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 616–627, 1998.

[12] J. J. Little and Z. Gu, "Video retrieval by spatial and temporal structure of trajectories," in *Proc. SPIE Storage and REtrieval for Media Databases*, San Jose, Calif, USA, 2001.

[13] S. W. Smoliar, J. D. Baker, T. Nakayama, and L. Wilcox, "Multimedia search: A authoring perspective," in *Proc. 1st International Workshop on Image Databases and Multimedia Search (IAPR-1996)*, pp. 1–8, Amsterdam, The Netherlands, August 1996.

[14] P. Over and R. Taban, "The TREC-2001 video track framework," in *Proc. 10th Text REtrieval Conference (TREC-10)*, Gaithersburg, Md, USA, November 2001.

[15] M. Flickher, H. Sawhney, W. Niblack, et al., "Query by image and video content: The QBIC system," *IEEE Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[16] J. Garofolo, C. Auzanne, and E. Voorhees, "The TREC spoken document retrieval track: A success story," in *Proc. 8th Text REtrieval Conference (TREC-8)*, E. Voorhees and D. Harman, Eds., vol. 500-246 of *NIST Special Publication*, pp. 107–130, Gaithersburg, Md, USA, November 1999.

[17] P. Aigrain, H. J. Zhang, and D. Petkovic, "Content-based representation and retrieval of visual media: A state-of-the-art review," *Multimedia Tools and Applications*, vol. 3, no. 3, pp. 179–202, 1996.

[18] J. R. Bach, C. Fuller, A. Gupta, et al., "Virage image search engine: An open framework for image management," in *IS&T/SPIE Conference on Storage and REtrieval for Still Image and Video Databases IV*, vol. 2670, pp. 76–87, San Jose, Calif, USA, February 1996.

[19] G. J. F. Jones, J. T. Foote, K. S. Jones, and S. J. Young, "Retrieving spoken documents by combining multiple index sources," in *Proc. Annual International ACM SIGIR Conference on Research and Development in Information REtrieval*, pp. 30–38, Zurich, Switzerland, August 1996.

[20] H. J. Zhang, P. Aigrain, and D. Petkovic, "Content-based representation and retrieval of visual media: A state-of-the-art review," *Multimedia Tools and Applications*, vol. 3, no. 3, pp. 179–202, 1996.

[21] S.-F. Chang, W. Chen, H. J. Meng, H. Sundaram, and D. Zhong, "VideoQ: An automated content based video search system using visual cues," in *Proc. ACM Multimedia*, pp. 313–324, ACM Press, Seattle, Wash, USA, November 1997.

[22] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428–440, 1999.

[23] H. Wactlar, M. Christel, Y. Gong, and A. Hauptmann, "Lessons learned from building a terabyte digital video library," *IEEE Computer*, vol. 32, no. 2, pp. 66–73, 1999.

[24] A. Del Bimbo, *Visual Information Retrieval*, Morgan Kaufmann, San Francisco, Calif, USA, 1999.

[25] A. Yoshitaka and T. Ichikawa, "A survey on content-based retrieval for multimedia databases," *IEEE Trans. Knowledge and Data Engineering*, vol. 11, no. 1, pp. 81–93, 1999.

[26] H. Greenspan, G. Dvir, and Y. Rubner, "Region correspondence for image matching via EMD flow," in *IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL-2000)*, pp. 27–31, Tel Aviv, Israel, June 2000.

[27] M. Petkovic and W. Jonker, "Content-based video retrieval by integrating spatio-temporal and stochastic recognition of events," in *Proc. IEEE International Workshop on Detection and Recognition of Events in Video (EVENT '01)*, Vancouver, BC, Canada, July 2001.

[28] Y. Gong, L. T. Sin, C. H. Chuan, H. J. Zhang, and M. Sakauchi, "Automatic parsing of TV soccer programs," in *Proc. IEEE International Conference on Multimedia Computing and Systems*, pp. 167–174, Washington, DC, May 1995.

[29] D. M. Gavrila, "The visual analysis of human movement: A survey," *Computer Vision and Image Understanding*, vol. 73, no. 1, pp. 82–98, 1999.

[30] R. Qian, N. Hearing, and I. Sezan, "A computational approach to semantic event detection," in *Proc. Computer Vision and Pattern Recognition*, vol. 1, pp. 200–206, Fort Collins, Colo, USA, June 1999.

[31] M. R. Naphade, T. Kristjansson, B. Frey, and T. S. Huang, "Probabilistic multimedia objects (multijects): A novel approach to video indexing and retrieval in multimedia systems," in *Proc. 5th IEEE International Conference on Image*

Processing, vol. 3, pp. 536–540, Chicago, Ill, USA, October 1998.

[32] P. Salembier and F. Marqués, "Region-based representations of image and video: segmentation tools for multimedia services," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1147–1169, 1999.

[33] X. He, W. Y. Ma, O. King, M. Li, and H. J. Zhang, "Learning and inferring a semantic space from users relevance feedback for image retrieval," in *Proc. ACM Multimedia*, France, December 2002.

[34] A. Hauptmann and M. Witbrock, "Informedia: News-on-demand multimedia information acquisition and retrieval," in *Intelligent Multimedia Information REtrieval*, chapter 10, pp. 215–240, MIT Press, Cambridge, Mass, USA, 1997.

[35] S. Srinivasan and D. Petkovic, "Phonetic confusion matrix based spoken document retrieval," in *Proc. 23rd Annual International ACM SIGIR Conference on Research and Development in Information REtrieval*, pp. 81–87, Athens, Greece, July 2000.

[36] M. G. Christel, M. A. Smith, C. R. Taylor, and D. B. Winkler, "Evolving video skims into useful multimedia abstractions," in *Proc. ACM CHI '98 Conference on Human Factors in Computing Systems*, pp. 171–178, Los Angeles, Calif, USA, April 1998.

[37] G. J. F. Jones, J. T. Foote, K. S. Jones, and S. J. Young, "Video mail retrieval: the effect of word spotting accuracy on precision," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, pp. 309–312, Detroit, Mich, USA, May 1995.

[38] E. M. Voorhees and D. Harman, "Overview of the sixth text retrieval conference (TREC-6)," *Information Processing and Management*, vol. 36, no. 1, pp. 3–35, 2000.

[39] L. He, A. Gupta, S. A. White, and J. Grudin, "Corporate deployment of on-demand video: Usage, benefits, and lessons," Tech. Rep. MSR-TR-98-62, Microsoft Research, Redmond, Wash, USA, November 1998.

[40] N. Omoigui, L. He, A. Gupta, J. Grudin, and E. Sanocki, "User benefits of non-linear time compression," in *Proc. ACM CHI '99 Conference on Human Factors in Computing Systems*, pp. 136–143, Pittsburgh, Pa, USA, May 1999.

[41] F. C. Li, A. Gupta, E. Sanocki, L. W. He, and Y. Rui, "Browsing digital video," in *Proc. ACM CHI '00 Conference on Human Factors in Computing Systems*, pp. 169–176, The Hague, The Netherlands, April 2000.

[42] S. Srinivasan, D. Ponceleon, A. Amir, and D. Petkovic, "What is in that video anyway? In search of better browsing," in *Proc. IEEE International Conference on Multimedia Computing and Systems*, pp. 388–392, Florence, Italy, June 1999.

[43] A. Amir, D. Ponceleon, D. Blanchard, B. Petkovic, S. Srinivasan, and G. Cohen, "Using audio time scale modification for video browsing," in *Proc. 33rd Hawaii International Conference on System Sciences (HICSS-33)*, vol. 3, Maui, Hawaii, USA, January 2000.

[44] D. Malah, "Time-domain algorithms for harmonic bandwidth reduction and time scaling of speech signals," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 27, no. 2, pp. 121–133, 1979.

[45] S. Srinivasan, D. Ponceleon, A. Amir, B. Blanchard, and D. Petkovic, *Web Engineering: Managing Diversity and Complexity in Web Application Development*, vol. 2016 of *Lecture Notes in Computer Science*, Springer-Verlag, Berlin, Germany, April 2001.

[46] A. Amir, G. Ashour, and S. Srinivasan, "Towards automatic real time preparation of on-line video proceedings for conference talks and presentations," in *Video Use in Office and Education, Proc. 34th Hawaii International Conference on System Sciences (HICSS-34)*, Maui, Hawaii, USA, January 2001.

[47] S. Chen, E. Eide, M. J. F. Gales, R. A. Gopinath, D. Kanevsky, and P. Olsen, "Automatic transcription of broadcast news," *Speech Communication*, vol. 37, no. 1-2, pp. 69–87, 2001.

[48] S. E. Johnson, P. Jourlin, G. L. Moore, K. S. Jones, and P. C. Woodland, "Spoken document retrieval for TREC-7 at Cambridge University," in *Proc. 7th Text REtrieval Conference (TREC-7)*, vol. 500-242 of *NIST Special Publication*, pp. 191–200, Gaithersburg, Md, USA, November 1998.

[49] A. Singhal, J. Coi, D. Hindle, D. Lewis, and F. Pereira, "AT&T at TREC-7," in *Proc. 7th Text REtrieval Conference (TREC-7)*, vol. 500–242 of *NIST Special Publication*, Gaithersburg, Md, USA, 1998.

[50] S. Srinivasan, D. Petkovic, D. Ponceleon, and M. Viswanathan, "Query expansion for imperfect speech: applications in distributed learning," in *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries (CBAIVL '00)*, Hilton Head Island, SC, USA, June 2000.

[51] S. E. Robertson, A. Walker, K. S. Jones, M. M. Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," in *Proc. 3rd Text REtrieval Conference (TREC-3)*, vol. 500-225 of *NIST Special Publication*, pp. 109–126, Gaithersburg, Md, USA, April 1995.

[52] R. Byrd and Y. Ravin, "Identifying and extracting relations in text," in *Proc. 4th International Conference on Applications of Natural Language to Information Systems (NLDB '99)*, pp. 149–154, Klagenfurt, Austria, June 1999.

[53] J. S. Justeson and S. M. Katz, "Technical terminology: some linguistic properties and an algorithm for identification in text," *Natural Language Engineering*, vol. 1, no. 1, pp. 9–27, 1995.

[54] K. Ng and V. Zue, "Phonetic recognition for spoken document retrieval," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, pp. 325–328, Seattle, Wash, USA, May 1998.

[55] M. Wechsler, E. Munteanu, and P. Schäuble, "New techniques for open vocabulary spoken document retrieval," in *Proc. Annual International ACM SIGIR Conference on Research and Development in Information REtrieval*, pp. 20–27, Melbourne, Australia, 1998.

[56] M. Witbrock and A. Hauptmann, "Using words and phonetic strings for efficient information retrieval from imperfectly transcribed spoken documents," in *Proc. 2nd ACM International Conference on Digital Libraries*, pp. 30–35, Philadelphia, Pa, USA, July 1997.

[57] P. Lawrence, "Hanging on the metaphone," *Computer Language*, vol. 7, no. 12, pp. 39–43, 1990.

[58] D. Gusfield, *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*, Cambridge University Press, Cambridge, UK, 1997.

[59] J. Allan, J. Carbonell, G. Doddington, J. P. Yamron, and Y. Yang, "Topic detection and tracking pilot study: final report," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, pp. 194–218, Lansdowne, Va, USA, February 1998.

[60] S. Dharanipragada, M. Franz, and S. Roukos, "Audio-indexing for broadcast news," in *Proc. 7th Text REtrieval Conference (TREC-7)*, vol. 500-242 of *NIST Special Publication*, pp. 63–67, Gaithersburg, Md, USA, 1999.

[61] M. A. Siegler, M. J. Witbrock, S. T. Slattery, K. Seymore, R. E. Jones, and A. Hauptmann, "Experiments in spoken document retrieval at CMU," in *Proc. 7th Text REtrieval Conference (TREC-7)*, vol. 500-242 of *NIS Special Publication*, pp. 264–270, Gaithersburg, Md, USA, November 1998.

[62] NIST Spoken Natural Language Processing Group, "Hub-4 broadcast news evaluation English test material," LDC catalog LDC2000S86, LDC, University of Pennsylvania, Philadelphia, Pa, USA, 1998.

[63] D. A. Van Leeuwen, W. Kraaij, and R. Ekkelenkamp, "Prediction of keyword spotting performance based on phonemic contents," in *Proc. ESCA ETRW Workshop: Accessing Information in Spoken Audio*, University of Cambridge, Cambridge, UK, 1999.

[64] A. Amir, A. Efrat, and S. Srinivasan, "Advances in phonetic word spotting," in *Proc. 10th International Conference on Information and Knowledge Management (ACM CIKM)*, pp. 580–582, Atlanta, Ga, USA, November 2001.

Arnon Amir is a research staff member at the IBM Almaden Research Center in California. He received his B.S. degree (magna cum laude) in electrical and computer engineering from the Ben-Gurion University, Israel in 1989, and the M.S. and Ph.D. degrees in computer science from the Technion-Israel Institute of Technology in 1992 and 1997, respectively. His research interests include computer vision, video and speech indexing and retrieval, eye gaze tracking, image and video segmentation, graph clustering, and spatial data structures. Dr. Amir has published many technical papers and holds several patents. He was awarded the 1997/98 Rothschild fellowship and is a member of the IEEE and of the ACM.

Savitha Srinivasan manages multimedia content distribution activities at IBM Almaden Research Center. Her group is responsible for multimedia information retrieval and content protection technologies. They are the founding members of copy protection technology currently deployed for DVD Audio/Video and have been top performers at the recent NIST-sponsored video retrieval task. Her research interests include video segmentation and semantic video retrieval with a focus on the application of speech recognition technologies to multimedia. She has published several papers on speech programming models and multimedia information retrieval. She is on the Scientific Advisory Board of a leading NSF multimedia school and is an Area Editor of multimedia in leading journals. She holds three patents related to the use of spelling in speech applications and the combination of speech recognition and audio analysis for information retrieval. Her current expertise extends into pragmatic aspects of multimedia such as digital rights management.

Alon Efrat received his Ph.D. in 1998 from the Computer Science Department at Tel Aviv University, and his M.S. in 1993 from the Technion Technological Institute in Israel. He then was a Post-Doctorate Research Assistant in the Computer Science Department at Stanford University and at IBM Almaden research center. Dr. Efrat joined the Computer Science Department at the University of Arizona, Tucson in 2000. His research interest includes algorithms in geometric settings, shape matching, and geometric optimization.