Multimedia Search for the Net

Mahesh Viswanathan, Fereydoun Maali, Alain Tritscher, Homayoon S.M. Beigi

Abstract— **We have devised a multimedia system that makes name-voice and name-face associations using speech, speaker, and face recognition. During the audio and video analysis phase, this system generates an approximate transcript, speaker boundaries, and speaker identities. A complementary retrieval engine and client station are used to search and retrieve digitized audio and video in response to queries.**

Keywords— **Multimedia indexing and retrieval, audio-visual analysis, speech recognition, speaker recognition**

I. Introduction

VIDEO content is being produced today at a prodigious rate. The last year alone has seen a rapid emergence of video web sites with television, movie and video content. This is in addition to the existing base of internet video projects from the major broadcast networks (ABC, CBS, CNN, FOX, and NBC). All of these TV broadcast organizations see the merit of converting their traditional analog and tape oriented data into digital means. There are also some sites (ReplayTV.com and TiVo.com) offering full-fledged digital TV via software that downloads user-specified, MPEG2-encoded TV programs every night over a telephone line onto a special digital "receiver" with a very large hard drive. Between RealNetworks, Microsoft and Apple, the installed base for downloadable, high-quality, very low-cost audio-video streaming players number over a 100 million.

Almost all news sites offer some means of access to the video or audio via plug-ins. What these news sites generally incorporate is some proprietary means of indexing the material. In some cases, manual mark-up will suffice. For the more consistent producer more cost-effective means are called for. Video for common internet consumption is offered as clips. Video is expensive to store and stream and these clips tend to be short. They are accessed by clicking on links that spell out the title of the clip or advertise its contents. Audio is offered in a similar manner.

Text on the web has grown dramatically over the last few years and is expected to grow at that rate. All major full-text search engines on the Web have made a large and growing body of information resources accessible within seconds by indexing close to the entire Web as it grows. But what about the new media such as image, audio, and video? To paraphrase Rudyard Kipling, text is just text, while audio and video consist of images, speech, music, text and other components racing by at several thousand to millions of bits per second. There is a vast ocean of information to contend with, and how do we extract all this information?

We describe a series of techniques, tools, and applications that facilitate the automatic analysis of audio (derived from a video clip or stream) towards the goal of automatically generating a multimedia index for later retrieval [1, 2]. This index retains the required markers to retrieve the relevant portions of the original video in response to a query. Note that we use audio and video interchangeably to indicate that we deal with media sources with integrated audio-video data, i.e., the two are tied together in a time-synchronous manner. Shot changes, cuts, fades, and other scene changes may pepper the video, but since the audio is recorded alongside the video they tell a common story. With a speech recognition engine dedicated to broadcast news, a generic speaker segmentation engine, a speaker identification and a face recognition subsystem working in close conjunction to each other, we show how such an indexing system can be built. We have also built a retrieval engine and a query interface to exploit it.

Our system uses a multimodal and multi-channel approach to the task of multimedia analysis for indexing and retrieval by extracting text, speaker, and face information from a audio-video stream. The oldest form of retrieval in the computer era is perhaps text retrieval referred to in the research literature rather presumptuously as "information retrieval" [3]. Still this ushered in the era of digital storage of documents. Image retrieval by content is a lot more complex, and to say the very least, cumbersome. Throw in moving pictures and speech and the need for elegant algorithms for automatic processing becomes pressing. Today, it is possible to obtain transcripts of audio clips, even if approximate, using the appropriate speech recognition system prepared for that domain. Generic speech recognition might do the trick, but the serious practitioner would call for a specialized one. Most news broadcasts in the US, and a larger percentage is mandated in the years to come, have closed captioning, but that still leaves out archival material and live, breaking news to be analyzed. The big advantage of speech recognition is that the approximate transcript is available as and when the news feed arrives and is immediately ready for further manipulation. Further, it lends itself to full-fledged automation on a twenty-four hour, seven-day basis. In addition to automatic speech recognition, we employ automatic speaker segmentation, speaker identification, and video-based face recognition. Speaker recognition yields rich marked up transcripts when adapted for use alongside speech recognition. Further it opens a whole new dimension for constraining search. Face recognition is used to validate audio-based speaker identification result via a

Mahesh Viswanathan is with the Human Language Technologies department, IBM Thomas J. Watson Research Center, Yorktown Heights, New York, USA. E-mail: maheshv@watson.ibm.com .

Fereydoun Maali is with the Signal Recognition Corporation, New York, USA. E-mail: maali-sigRec@worldnet.att.com .

Alain Tritschler is with Envivo Corporation, Rennes, France. E-mail atritschler@envivo.com .

Homayoon Beigi is with Internet Server Connections, Inc., White Plains, New York, USA. E-mail: beigi@internetserver.com .

decision integration scheme. We have used a set of CNN broadcast news programs as our audio and video source.

A fully functioning version of the system described in this paper exists today. When live news is fed into the sound card of a computer loaded with our system, our application performs multiple operations on the incident audio. (All of our audio modules require that the input audio signal is converted into multi-dimensional feature vectors. This is a canonical step in the processing of time-varying signals. This is referred to as front-end[1] processing.) The transcription engine generates string after string of text onto the application whiteboard. The speaker segmentation engine detects any change in speakers by listening in on the same audio and produces visual line breaks to delineate speakers on the user interface. A speaker identifier uses the input signal and the demarcated speaker segment, compares the audio signal against its database of interesting speakers (using name-voice association), picks the closest matching speaker's name and renders it on the whiteboard next to the transcribed text. The face recognition engine uses the same speaker segmentation result to find the "speaker" (using name-face association) employing face identification algorithms that can match facial features against a databases of faces and features [4]. However, the video input is processed separately on another machine (it is too compute-intensive to be on same computer as the rest of our audio-processing modules) using the video input from the same source. These identified names are also etched alongside the transcribed text on the same user interface.

The essence of multimedia retrieval is in delivering the requested information as soon as possible to the user [5, 6, 7]. We have built a retrieval system that can retrieve video or audio clips provided that the video is encoded alongside the audio analysis step. Encoding video entails digitizing the analog stream from a news feed or a VCR playing a tape into a compressed stream such as MPEG. There are several tools and toolkits available to manipulate MPEG, our principle interest in being able to play and stop playing at some point within the clip on demand. In response to a query, the retrieval engine selects the most appropriate portion of the conveniently apportioned transcribed text for display to the user. This apportionment is part of the indexing process which breaks up the continuously flowing transcribed text into manageable chunks of a pre-determined size, say 30 seconds worth of text. The name-face association provided in our application is produced artificially for visual impact. In order to increase the visual appeal we have separately captured snapshots of speakers in our database of speakers and these pictures are loaded on the application whiteboard when a speaker is identified. Likewise for the face recognized individuals. The figures in this paper feature these pictures.

A typical indexing session might run as follows. The audio feed is delivered to the three engines cited above. The outputs of the three engines are captured by the application and in the program memory. When the audio terminates or the program is stopped, the transcribed text is converted into a series of conveniently sized text chunks which is unrelated to the speaker segment information at this point. An accurate time-stamp is also obtained for each word transcribed which helps to relate the position of each word back into the input video. In addition, the speaker segment boundaries are captured along with the speaker names assigned to each segment. These are variable sized sections of the audio and we only record the starting and ending points of the segments, the labels (names) assigned, and a score which denotes how well this audio segment matches the speaker in our database. Since these matching functions are based on statistics, we cannot have an absolute match but instead we have a set of names and scores from best to worst. We trim this list to retain the top six speakers and scores. Using a completely different scheme, a name-face association table is recorded by the face recognition system, once again with a list of matches from best to worst. Each match consists of a name (recorded during face enrollment) and a score (determined during the processing pass). The architecture and flow diagram is in Fig. 1.

The start and end times of each of text chunks are recorded as these later become are the units of video retrieval. After morphological analysis (a form of linguistic signal processing in which nouns are decomposed into their roots along with a tag to indicate the plural form and verbs are decomposed into units designating person, tense and mood, along with the root of the verb), statistics are extracted from each of these "documents" as required by the Okapi equation and recorded, viz., term frequency and inverse document frequency [8]. The speaker index is a list of speaker segments with an assigned label (best matched), a score, the starting and ending time of the segment. The face index is also a name, a score over the same speaker segment. These two results are "fused" to come with a consensus name and score that best reflects the confidence each scheme has on the label it has assigned. The time involved in generating the various index files (text and speaker) is around 1–2% of the time required in transcription.

We expect our system to be used either as a standalone application or on the web as a multimedia search engine. In either case, the user can query the system by textual queries to retrieve videos, audios, and text documents. Hence, if former US President Bill Clinton is the subject of interest then the retrieved material should contain those words. In addition, the system can also retrieve on the basis of speaker queries - so if the requested speaker is Bill Clinton, then the retrieved videos must include a segment spoken by Bill Clinton. A combination search is also possible which places limits on the subject material ("the internet") and is restricted by speaker of interest ("Al Gore"). This combination search will yield portions of videos on this subject but only if spoken by the former US Vice-President Al Gore. Hence, our system has several applications including:

- a multimedia search engine
- a news video viewer which interactively provides clips of

[1] shown as FE in Fig. 1

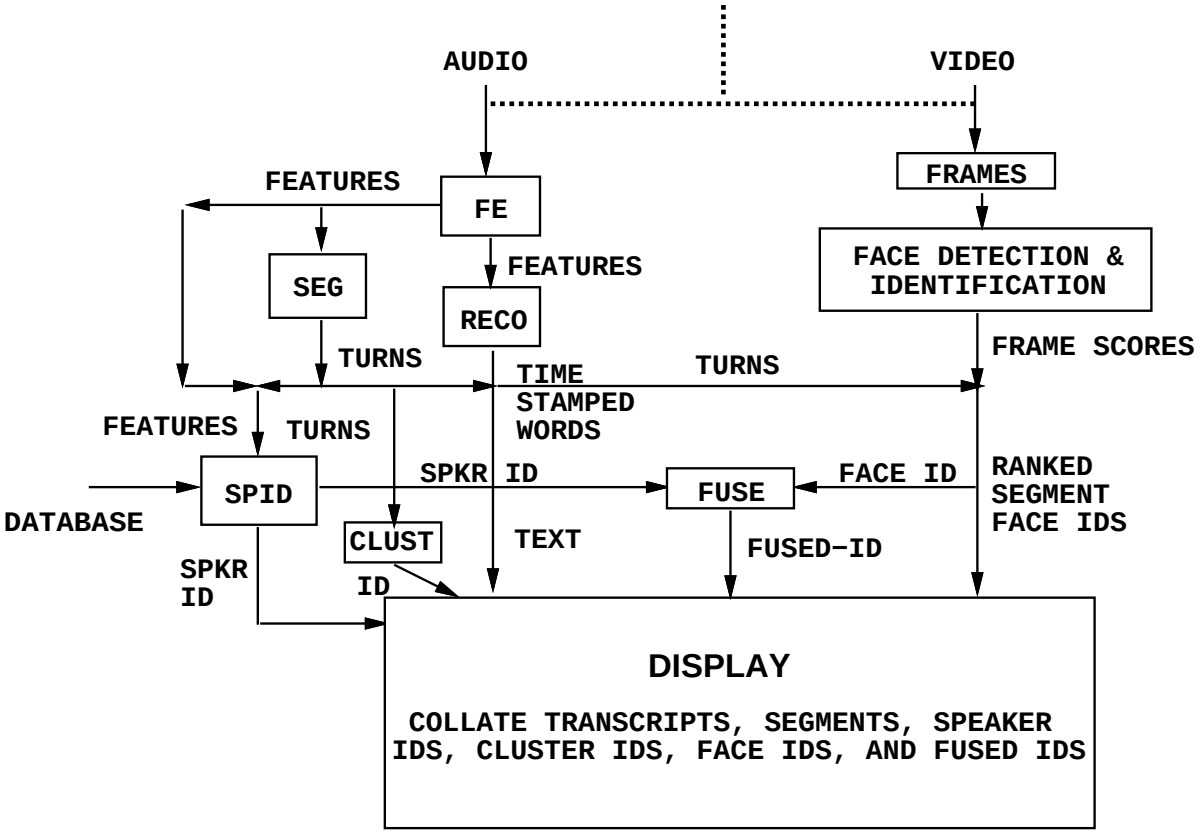


Fig. 1. The architecture of our audio and video analysis system.

user interest
• a system that processes news audio in real-time providing approximate textual transcripts while citing the speakers
• a system for news alerts by tracking certain news items and certain speakers of choice

II. Speech Transcription

Speech recognition systems are typically guided by a vocabulary, a language model and set of pronunciations for each word in the vocabulary. A vocabulary is a set of words that is used by the recognizer to translate speech to text. The recognizer is therefore considered to be decoding the audio to produce text. As part of the decoding process, the recognizer matches the acoustics from the speech input to words in the vocabulary. Therefore, the vocabulary defines and limits what words can be transcribed. If a word that is not in the vocabulary is to be recognized, it must first be added to the vocabulary. Hence, our earlier reference to a specialized speech recognition system. A system built for broadcast news will use only broadcast news audio and news wire sources and as such is unlikely to contain specialized words from the medical profession or legal profession in vast numbers. A language model is a domain-specific (say, office memo dictation, broadcast news, legal, or radiology) database of sequences of words in the vocabulary. A set of probabilities of the words occurring in a specific order is also required. These tend to act as tie-breakers or even boosters of words more likely in English usage in that particular context (within a domain) even if on the basis of pure acoustics the words sound the same. For instance, two words "right" and "write" may sound the same as far as the acoustics go, but the language model would break in favor of "write" if the preceding words are, "I have a letter to". The output of the recognizer will be biased towards the high probability word sequences when the language model is operative. Correct decoding is therefore a function of whether the user speaks a sequence of words that have high probability within the language model. Which is why when the user speaks an unusual sequence of words, the decoder performance will degrade. Word recognition is based entirely on their pronunciation, the phonetic representations of the word [9].

The speech recognition system described above was tested on the 1997 Hub4 evaluation test set which consists of three hours of broadcast news [10]. The overall correct recognition rate was 70.3% over a data set with various speech conditions including prepared, spontaneous, and low-fidelity speech, speech with background noise, speech with background music, and speech collected from those without a North-American accent. Clearly, the best performance was for prepared speech (a news anchor in a studio setting) at 77.8%, while at the other end of the spectrum for low-fidelity speech it was 60.4%. The numbers above reflect

the real-time performance values for the engine. Current versions perform roughly 15% better across the board.

III. Audio-based Speaker Segmentation

Speaker segmentation is the process of carving up an audio stream into homogeneous sections. Matching audio clips to name the speaker involves computing a match-score in a certain feature-space between a database of labeled voice-prints and a test segment of audio. The database of voice-prints can be captured in a controlled environment by isolating the speaker's utterances, collecting them into a disk file, and then submitting it to a separate application that "enrolls" these speakers into the database. In our scenario, the audio is from a broadcast news video and therefore is first broken up into sections corresponding to individual speakers segments. If we are able to isolate a segment on the fly we can match it against our database to determine its label, i.e., if it is one of our speakers of interest.

The scheme that performs this task with about a 80% accuracy and to within half a second on either side of the real segment boundary, is BIC-based segmentation [11]. The BIC-based segmentation engine uses the Bayesian Information Criterion to partition the feature vectors produced by the front-end. We view this as a two-class classification problem with the objective of determining whether N consecutive audio frames (1 audio frame = 10 ms of audio) constitute a single homogeneous window, W, or two such windows, W_1 and W_2, with a boundary (or turn) occurring at the ith frame. We build two models to detect whether a speaker change transpired within a window of N frames. One represents the entire window, W, by a Gaussian characterized by $\{\mu, \Sigma\}$; and a second that represents the window upto the ith frame, W_1, with one Gaussian, $\{\mu_1, \Sigma_1\}$, and the remaining part, W_2, with a second Gaussian, $\{\mu_2, \Sigma_2\}$, assuming independent but not uncorrelated feature elements.

$$\begin{aligned}\Delta BIC(i) &= -\frac{N}{2}log|\Sigma| + \frac{N_1}{2}log|\Sigma_1| + \frac{N_2}{2}log|\Sigma_2| \\ &\quad + \frac{\lambda}{2}(d + \frac{d(d+1)}{2})logN\end{aligned}$$

where d is the dimension of the cepstral vector; $N_1 = i$ is the number of frames in W_1, $N_2 = (N - i)$ is the number of frames of the second part; and lambda (= 1.3) is a penalty function. Taking the penalty into account, $\Delta BIC < 0$ implies that the model splitting the window into two Gaussians is more likely than the model representing the entire window with only a single Gaussian. Within the window the i where the largest negative ΔBIC is computed is considered the frame where the speaker change occurred.

IV. Audio-based Speaker Identification

Speaker recognition has two principal components, speaker identification and speaker verification [12]. Our engine is both text and language independent which is essential for live audio indexing of broadcast material. Let us first take up speaker enrollment - the process of adding new speakers of interest into a speaker database.

Individual speakers are modeled as a mixture of Gaussians represented by the mean, covariance, and number of distributions within that model. A k-means clusterer generates the speaker models from their sample voice-prints (about 30 seconds worth). A binary tree is constructed bottom-up with speaker models as the leaves of the tree. At each higher node, the closest two models are merged using a similarity measure [13]. Each sub-tree of this tree therefore contains the closest relatives of any leaf-node speaker - its confusable or cohort set.

As the speaker segmentation engine produces new segment boundaries, the first eight seconds of each new segment is passed to the speaker recognition engine for identification followed by verification. (Segments smaller than eight seconds are dismissed by the speaker identification engine without consideration and assigned an "Inconclusive" label.) The following is an explanation of how identification is done. Let $\{M_i \mid i = 1..I\}$ denote the models pertaining to the enrollees. Each model M_i can have N_i distributions associated with it. Let ω_{ij} refer to the jth distribution of model i. Also, let $\{\vec{x}_t \mid t = 1..T\}$ denote the frames constituting the test utterance, $\mathbf{z}$, whose label is sought. During run-time, a test utterance $\mathbf{z}$ is identified with model q according to:

$$\text{assign } \mathbf{z} \longrightarrow \mathrm{M_q} \text{ iff } \mathrm{D_q} = \min_{\mathrm{i=1..I}} [\mathrm{D_i}], \text{ where}$$

$$D_i = \sum_{t=1}^{T} d(i,t), \; i = 1..I, \text{ and}$$

$$d(i,t) = -log[\sum_{j=1}^{N_i} P(\omega_{ij})\; p(\vec{x}_t|\omega_{ij})],$$

where $P(\omega_{ij})$ is the prior of the jth distribution of model i, and $p(\vec{x}_t|\omega_{ij})$ is the conditional pdf of utterance conditioned on the jth component of model i. A Normal representation for $p(\vec{x}_t|\omega_{ij})$ is used. Each class assignment is accompanied by a score which expresses the degree of confidence in the match. For model i, score = $\mathrm{D_i} \times \mathrm{T}$, where T denotes the number of frames in the respective utterance. Hence, each model in the database is accompanied by a set of rankings as identification result.

Verification follows identification as a separate but essential step. A model - akin to a speaker's enrollment model - is generated from the test utterance for this step. The identified speaker is verified when this test model is closest to the speaker's enrollment model when compared to the rest of its relatives in the cohort set. The combined performance of the speaker segmentation and identification components of our system are shown in Table I. 75 segments in all were submitted for identification. 70 were correctly identified and verified. Of the five mis-identifications, four were upheld, and only one was erroneously rejected.

TABLE I
MULTIPLE-SPEAKER SEGMENTATION AND IDENTIFICATION RESULTS FROM PROCESSING A SINGLE AUDIO FILE.

Speaker segments	104
Segments reported	84/104 (80.8%)
Segments missed	25/104 (23%)
Oversegmentations	5/104 (4.8%)
Identified	70/75 (93.3%)
Inconclusive	9/84 (10.7%)
Verified (from identified)	70/70
Verified (from mis-identified)	4/5
Overall Verification	74/75 (98.7%)

V. Another Speaker Identifier Using BIC Clustering

We made two enhancements to our speaker identification using the tools already at our command. As much as BIC-based segmentation is a technique to divide an audio stream into acoustically similar segments, it is also a technique that can splice together segments (or cluster) if the criterion is reversed [11]. That is, if the computed ΔBIC between each new segment and any extant cluster is positive, then these two have similar acoustics (and therefore are from the same speaker). If there are multiple clusters with positive ΔBIC the new segment is assigned to the one with the largest ΔBIC. If it is negative with every prior cluster, the new segment seeds a new cluster. Since BIC clustering requires on average three seconds to determine if two segments can be clustered together, we can re-assign some of the segments labeled "Inconclusive" by the speaker identification technique. (Segments shorter than three seconds are assigned a cluster id of 0 which is synonymous with the "Inconclusive" label.) It is important to note that two contiguous segments cannot belong to the same speaker because if this were true they wouldn't result in two different segments in the first place.

The BIC clustering engine assigns unique cluster labels to new segments as they are created. This engine works concurrently with the speaker identification engine. For every segment both its speaker label and BIC-cluster label are recorded for the duration of the audio. Once the audio terminates, if every instance of speaker label "John Doe" is assigned cluster label 26, then all the segments labeled "Inconclusive" and cluster label 26 are re-assigned to "John Doe".

BIC clustering tends to aid in improving overall speaker identification performance. Longer segments used in speaker labeling lead to better identification results, but this also increases the number of segments that are marked "Inconclusive." Remember also that BIC segmentation tends to generate more segments than warranted - up to 6%. To address these shortcomings, we integrated the results of speaker identification with that of BIC based speaker classification. We construct a scattergram with BIC cluster ids and speaker labels on the two axes. We view accumulation in any cell of the scattergram as evidence of conformity. Fig. 2 is a scattergram for a 30-minute test video sequence with 12 scored speakers using a 43-speaker database. Note the conformity in the cells corresponding to speaker 12–cluster id 5, and speaker 13–cluster 12. We deduce that the "Inconclusive" segments with cluster ids 5 and 12 belong to speakers 12 and 13 respectively. (We have attested this by manual inspection.) Therefore, all the "Inconclusive" segments with cluster id 5 can be re-labeled as speaker 12. Re-assigning the "Inconclusive" labels to the rightful owners only works when the accumulation in any cell is greater than three. This we have ascertained by experimentation. Hence, the above speakers are the only ones re-labeled in this particular run of the test depicted in the figure.

VI. On-the-fly Speaker Enrollment

BIC clustering also aids in partial automation of the speaker enrollment process. The clustering engine is used to process the input audio and all segments assigned to the same cluster id are labeled via a pop-up in which the user fills out the name of the speaker in the audio. Simultaneously, the raw PCM audio is written out to a disk file, one per cluster, with the user-assigned label as file name. When enough speakers of interest have been labeled (or by overt user action), the gathered PCM audios are submitted for speaker enrollment. The newly labeled speakers are all now part of the speaker database. This process is also available for speaker emendations to correct mis-labeled speakers when the system is processing any audio.

VII. Video-based Name-Face Association

Speaker identification and BIC-clustering are both audio-based techniques whose recognition performance degrades when faced with audio signal degradations. However, by integrating decisions from a completely different (and orthogonal) scheme, such as face recognition for labeling speakers, we overcome this disadvantage.

The general steps in face-name association are face tracking and face identification. The input video stream is analyzed frame-by-frame to first segment the image by partitioning it into face and non-face regions and then isolating the faces from each other. A face detection process is initiated to segment the image within a single video frame. To avoid employing the expensive detection operation on each frame, we attempt to "track" the face temporally in each subsequent frame after it is first detected. Face detection is required in any frame only when tracking fails and cannot be maintained.

Within each face a number of landmarks are located. These landmarks are distinctive points on a human face. Face identification is based on landmark recognition. A face is assigned to any one of the number of prototype face classes when its landmarks exhibit the highest comparative similarity to the constituent landmarks of a given prototype face. For face recognition we are using a package whose details appear in [14].

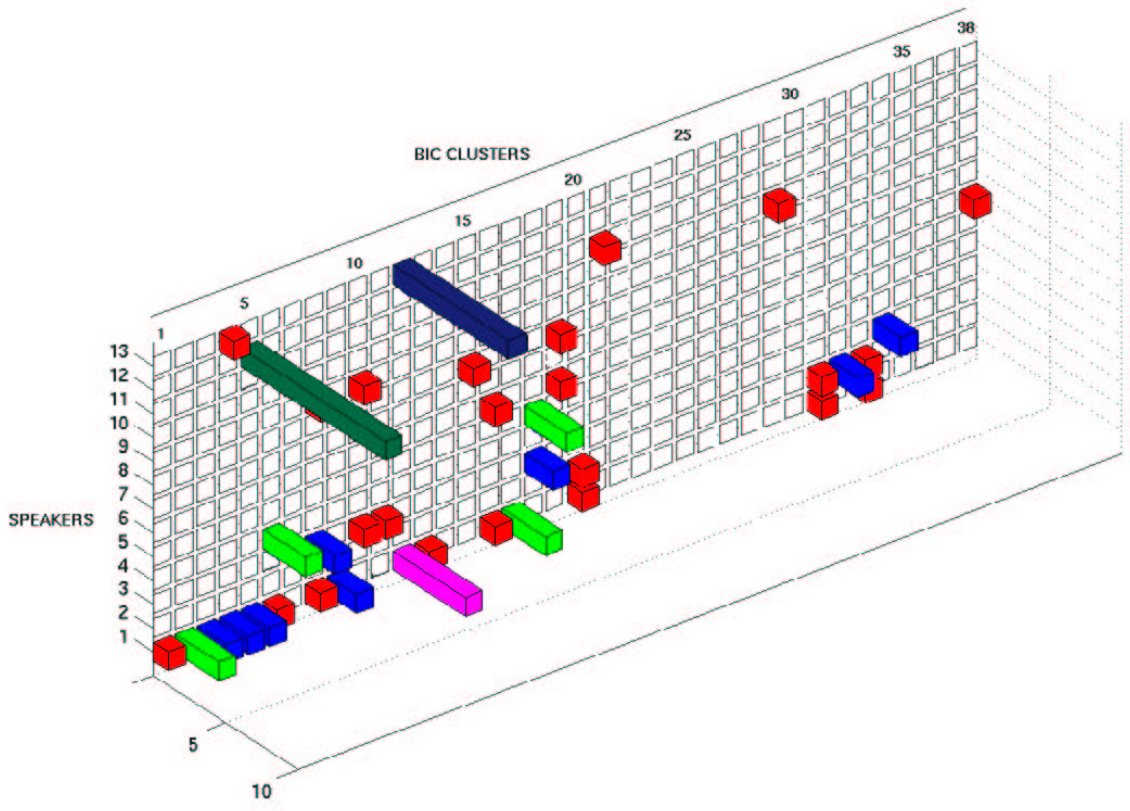


Fig. 2. Scattergram showing conformity between SVAPI and BIC based speaker identification schemes. This is generated on a 30-minute test audio with 12 labeled speakers from a database of 43. Note that Speaker 1 is really "Inconclusive."

VIII. Decision Integration - Audio and Video Channels

The audio-based speaker identification yields a single set of ranked identities for each audio segment with concomitant match scores. Corresponding to each audio segment, however, we have multiple video frames with one set of ranked identities for each video frame. Each identity is also accompanied by a confidence measure expressed as a score.

The results for N video frames that make up the equivalent of a single speaker segment must be abstracted to yield a single set of ranked identities for the entire video (speaker) segment. For the purposes of this analysis, the bounds of the video segments are identical to that of the audio speaker segments. The steps involved are: 1) finding the most frequent face identity (the statistical mode) at each rank position across all the video frames corresponding to the audio speaker segment; and 2) computing the median score for that rank and assigning it to the most frequent face identity. We now have one set of ranked identities for the entire video segment delineated by the audio segment. In the next step, the audio speaker scores are scaled to a 0-1 range and normalized by the standard deviation of the scores of the ranked identities for each segment. This is repeated for the video segment scores. This operation brings the video (face) and the audio (speaker) scores into more or less the same dynamic range suitable for subsequent integration.

Our decision fusion scheme is based on the linear combination of the audio and the video class assignments. The weights assigned to the audio and the video scores influence their respective scores in the ultimate outcome. One approach is to use fixed weights, say 0.5, for each. Another approach is to derive the weights from the data itself. The rest of the formulation described here uses this latter approach. A snapshot of our application demonstrating decision fusion and subsequent labeling is in Fig. 3.

Let $\{(rank_r, audioScore_r) \mid r = 1..maxRank\}$ denote the scores for ranked identities for audio speaker class assignments in a rank-score coordinate system, where $rank_r$ represents the rank and $audioScore_r$, the audio score of the rth rank. In the same manner, let $\{(rank_r, videoScore_r) \mid r = 1..maxRank\}$ denote the corresponding data set for the video identities. Both audio-based and video-based vary monotonically along the rank axis (Fig. 4). We impose a linear variation on the rank-score data by: (1) removing the outliers using the Hough transform; and (2)

Fig. 3. Application snapshot showing results of speaker and face identification. The audio id is on the left, the video id is on the right and the fused id is shown in the main panel alongside the transcribed segment. The "Inconclusive" segment is too short (less than three seconds) to be altered by BIC clustering.

fitting the surviving points set to a line using the least mean squares error method. Thereby the ranked identities output by the audio and video identification schemes are reduced to two straight lines defined by $audioScore = m_1.rank + b_1$ & $videoScore = m_2.rank + b_2$. The line with higher slope clearly conveys more discriminative information. The normalized slopes of the two lines are used as the weights when combining the scores from the audio-based and video-based speaker analyses.

With w_1 and w_2 representing audio and the video channels respectively we compute the fused score, FS_k, for each speaker as follows:

$$FS_k = w_1\left[m_1.rank_k + b_1\right] + w_2\left[m_2.rank_k) + b_2\right],$$
$$\text{where } w_1 = \frac{m_1}{m_1 + m_2} \text{ and } w_2 = \frac{m_2}{m_1 + m_2},$$

and $rank_k$ is the rank for speaker k. The above expression is computed over the collection of audio and video speaker identities derived from the last step, and later sorted by score to obtain a new set of fused ranked identities. These fused identities are displayed to the user as the identified speaker result, and recorded for subsequent use in speaker indexing.

The above fusion scheme was applied to a broadcast news video clip with eight speakers. Table II illustrates the result and the impact of multi-channel decision fusion compared with audio-only and video-only speaker identification when applied to one segment of a three-minute broadcast news video clip with eight speakers. The true speaker may not be identified on either channel in the top positions. It is the integrated decision that matters the most. (The converse is also true. High scoring but erroneous identifications from either channel can strongly influence the integrated decision. Further experimentation is required to take extra high scores into consideration during fusion.)

IX. Retrieval

We developed a retrieval engine that uses the indexes built during audio and video analysis to find the most relevant portions of stored video broadcasts. The user's query can include both text and a speaker. In our implementation, we first run the textual query through the engine to obtain a set of candidate documents to be retrieved (say, 50), and then apply the speaker constraint. (If a requested speaker does not exist in our speaker database and is hence not in our index, we just ignore it and return just the top

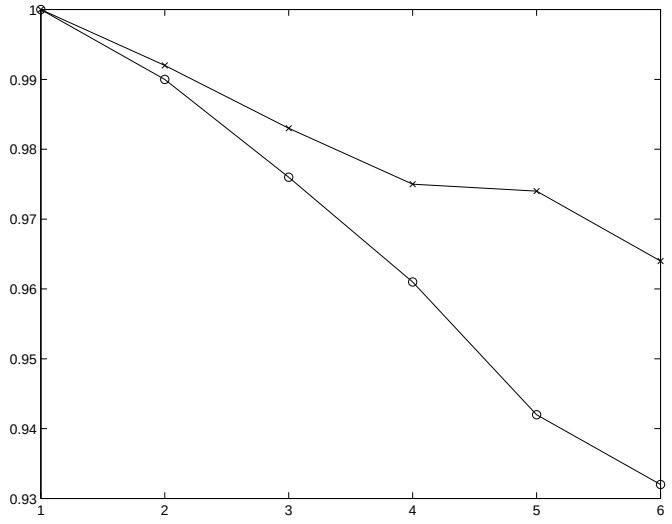


Fig. 4. Ranked id scores for audio (upper) and video (lower) for one audio segment with six rankings. See Table II for actual values.

TABLE II

FUSED RESULTS FOR SIX TOP RANKED SPEAKERS. GRAPH IN FIG. 4.

Rank	Audio		Video		Fused	
	Name	Score	Name	Score	Name	Score
1	JM	1.000	JM	1.000	JM	1.000
2	UM	0.992	GB	0.990	GB	0.987
3	GB	0.983	JW	0.976	SF	0.953
4	UF	0.975	AK	0.961	OH	0.953
5	SF	0.974	OH	0.942	UM	0.496
6	OH	0.964	SF	0.932	JW	0.488

N responses to the query from the text documents derived from the transcribed audio.) The following version of the Okapi formula, for computing the matching score, S, between a document d and a query q is used in determining the relevance between a query and the collection of text chunks that form the document database.

$$S(d,q) = \sum_{k=1}^{Q} c_q(q_k) \frac{c_d(q_k)}{\alpha_1 + \alpha_2 \frac{l_d}{\bar{l}} + c_d(q_k)} idf(q_k),$$

$$\text{where } idf(q_k) = log(\frac{N - n(q_k) + 0.5}{n(q_k) + 0.5}).$$

Here, q_k is the kth term in the query, Q is the number of terms in the query, $c_q(q_k)$ and $c_d(q_k)$ are the counts of the kth term in the query and document respectively, l_d is the length of the document, and $\bar{l}$ is the average length of the documents in the collection. $idf(q_k)$ is the inverse document frequency, where N is the total number of documents, $n(q_k)$ is the number of documents that contain the term q_k, and for unigrams, $\alpha_1 = 0.5$ and $\alpha_2 = 1.5$. The idf is pre-calculated and stored as are most of the elements of the scoring function except for query-related items.

One common ground to find an intersection between documents from the text-based search and the audio segments from the speaker-based search are the portions of audio (video) which overlap. This is recorded in their respective indexes as the start and end times of individual text chunks and audio segments. The algorithm to compute the combined score, C_s, from the two individual text and speaker searches is as follows. For each chunk from the text search, run down all the segments corresponding to the requested speaker computing time overlaps given by: $C_s = (c_s + (\lambda * s_s)) * (o_f)$, where c_s is the score for the retrieved document from the text-based search, s_s is the score for the speaker segment, and o_f is a fraction ($0 < o_f < 1$) that specifies by how much the speaker segment overlaps with the result of the text-based search. λ is a factor which handicaps s_s based on the confidence in the speaker scores. Currently, a λ of 0.75 is used. The resulting combined scores are sorted and normalized. A snapshot of our user interface displaying a retrieval result is shown in Fig. 5.

X. EXPERIMENTS

We digitized five hours of broadcast news video from VHS video tapes into disk files in a first pass. This is used solely for presentation during retrieval. We used two 30-minute segments as test segments to obtain the results presented here. We ran separate tests with 20 multi-word, text-only queries, 10 speaker-only queries, and 10 combined text-speaker queries. The top N performance is shown below as scored by manual count. In the combined search, errors are both due to speaker mis-identifications and irrelevant documents being retrieved. Sample text-speaker combination queries include "Mike Boorda-John McCain" and "Boris Yeltsin-Natalie Allen". All of these results are summarized in Table III.

TABLE III

COMBINED TEXT-SPEAKER RETRIEVAL PERFORMANCE FOR 20 TEXT-ONLY, 10 SPEAKER-ONLY, AND 10 TEXT-AND-SPEAKER QUERIES.

Search	Relevant/Retrieved
Text-only Top 5	198/200 (99%)
Text-only Top 25	143/200 (72%)
Speaker-only	77/99 (78%)
Combined Top 5	51/62 (82%)
Combined Top 10	78/93 (84%)

XI. CONCLUSIONS

Our system analyzes broadcast news audio to generate approximate transcripts, and to produce speaker labels or name-voice associations using two complementary speaker classification schemes. It also uses face recognition to validate speaker labels using name-face association. The generated text and speaker labels are used to build indexes for multimedia retrieval based on co-occurrence of relevant subject and speaker information. We have also built a retrieval engine and retrieval client station to illustrate our techniques. The successful experimental results demonstrate the feasibility and effectiveness of multimedia search

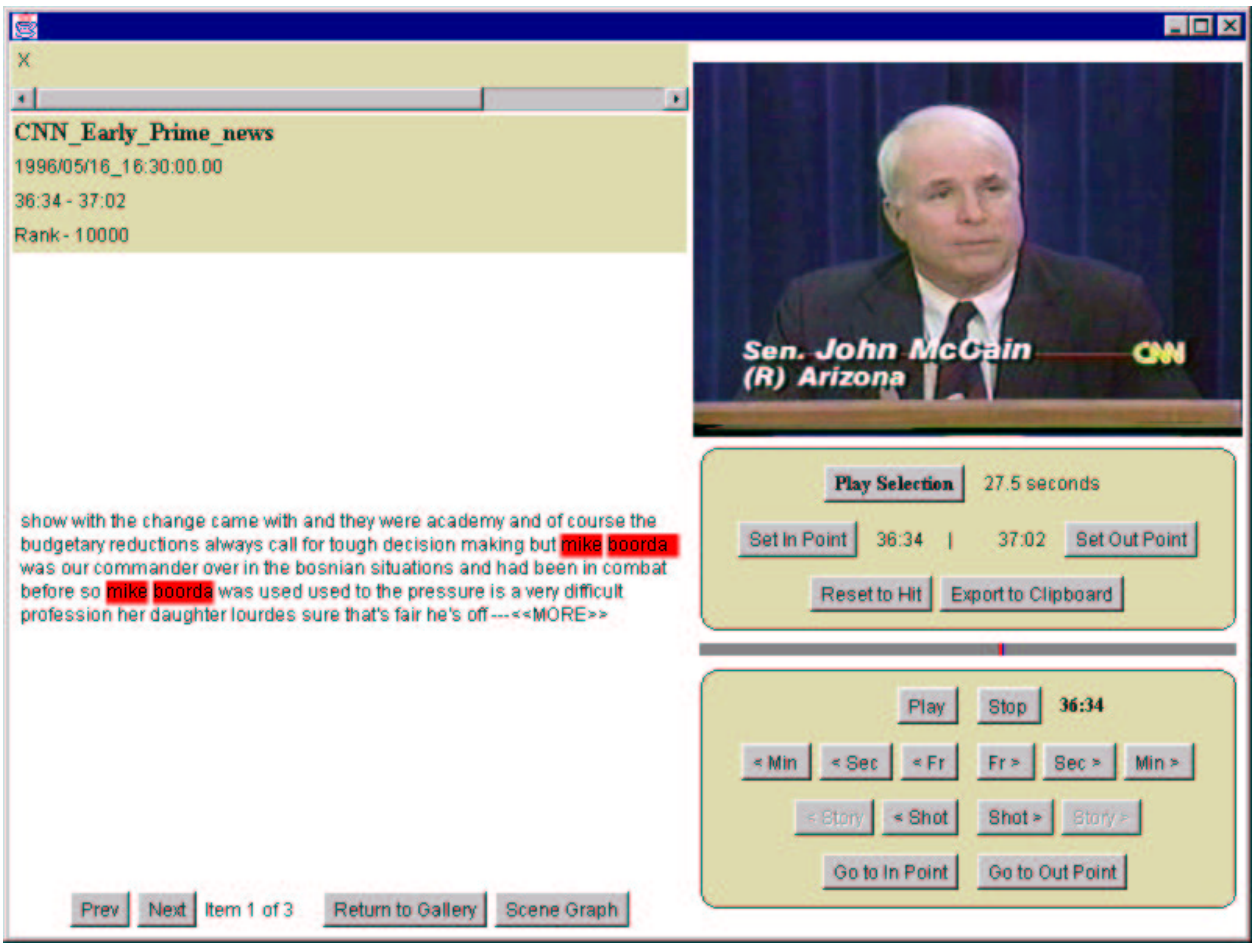


Fig. 5. The retrieved section for text query "Mike Boorda" and speaker "John McCain" is presented. This result corresponds to the top (most relevant) result for this query. Pressing "Play Selection" enables video playback of the retrieved segment.

and retrieval. Multi-channel and multi-sensory analysis is essential to extract meaning and value from audio and video. We have also demonstrated the power and applicability of decision integration to enhance speaker identification using face recognition. We will continue our research towards improving the overall system performance as well as analyze decision fusion further.

References

[1] M. Viswanathan, H.S.M. Beigi, S. Dharanipragada, and A. Tritschler, "Retrieval from Spoken Documents Using Content And Speaker Information," *Proc., Intl. Conf. on Document Analysis and Retrieval (ICDAR99)*, Bangalore, India, 1999, 567–572.

[2] E. Wold, T. Blum, and D. Keislar, "Content-based Classification, Search, and Retrieval of Audio," *IEEE Multimedia*, Volume 3, Number 3, Fall 1996, 27–36.

[3] G. Salton, *Automatic Text Processing*, Addison-Wesley, 1989.

[4] S. Satoh, Y. Nakamura, and T. Kanade, "Name-It: Naming and Detecting Faces in News Videos," *IEEE Multimedia*, Volume 6, Number 1, January-March 1999, 22–35.

[5] J. Hirschberg, S. Whittaker, D, Hindle, F. Periera, and A. Singhal, "Finding Information in Audio: A New Paradigm for Audio Browsing and Retrieval," *Proc., ESCA Tutorial and Research Workshop*, Cambridge, United Kingdom, April 1999.

[6] W. Grosky, "Pushing Streaming Video – Indexing Video Archives," *IEEE Multimedia*, October-December 1997, 7–8.

[7] J. Makhoul, F. Kubala, T. Leek, D. Liu, L. Nguyen, R. Schwartz, and A. Srivastava, "Speech and Language Technologies for Audio Indexing and Retrieval," *Proc. of the IEEE*, Vol. 88, No. 8, August 2000, 1338–1353.

[8] S.E. Robertson, S. Walker, K. Sparck-Jones, M.M Hancock-Beaulieu, and M. Gatford, "Okapi at TREC-3," *Proc., Third Text Retrieval Conf. (TREC-3)*, D.K. Harman, editor, NIST Special Publication 500–226, Gaithersburg, Maryland, 1995, 109–126.

[9] L. Rabiner and B.H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.

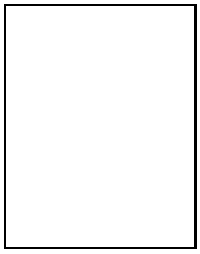
[10] D. Pallet, "Overview of the 1997 DARPA Speech Recognition Workshop," *Proc., DARPA Speech Recognition Workshop*, Chantilly, Virginia, 1997, 1–2.

[11] A. Trischler and R.A. Gopinath, "Improved Speaker Segmentation and Segments Clustering Using the Bayesian Information Criterion," *Proc., EuroSpeech99*, Budapest, Hungary, 1999, 679–682.

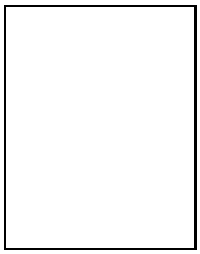
[12] H.S.M. Beigi, S. Maes, U.V. Chaudhari and J.S. Sorenson, "IBM Model-Based and Frame-By-Frame Speaker-Recognition," *Proc., Speaker Recognition and its Commercial and Forensic Applications*, Avignon, France, 1998.

[13] H.S.M. Beigi, S. Maes and J.S. Sorenson, "A Distance Measure Between Collections of Distributions and Its Application to Speaker Recognition," *Proc., Intl. Conf on Acoustics, Speech, and Signal Processing*, Seattle, Washington, 1998, 753–756.

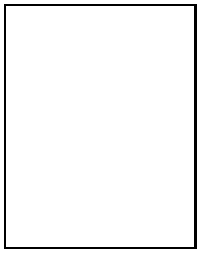
[14] A. Senior, "Recognizing Faces in Broadcast Video," *Proc. IEEE Intl. Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems*, Kerkyra, Greece, 1999, 105-110.



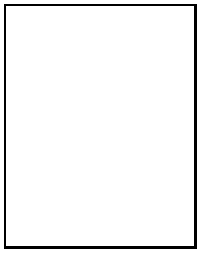
Mahesh Viswanathan is a Research Staff Member with the Human Language Technologies department at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York. His research interests include text-to-speech synthesis, multimedia indexing and retrieval, information retrieval, speech recognition, and document analysis. Viswanathan has a BS in Physics from Loyola College, Madras, India, BS in Electrical Engineering from the Indian Institute of Science, Bangalore, India, MS in Electrical and Computer Engineering from San Diego State University, San Diego, California, and PhD in Computer and Systems Engineering from Rensselaer Polytechnic Institute, Troy, New York. He has worked for IBM since 1991 in research, advanced technology, and product management and is a senior member of the IEEE.



Fereydoun Maali is President of Signal Recognition Corporation, a Manhattan-based software engineering firm specializing in image processing and industrial vision. He has acted as consultant in image processing and related areas to a number of corporations including IBM, Nynex (now Verizon), United Technologies, RVSI, and Universal Instruments. He was earlier involved in research and development in image processing and robotic vision for Robotic Vision Systems, New York, and Vickers (Joyce-Loebl Div.), UK. Until 1979 he was a Commander in the Imperial Iranian Navy. He has a DIC, MSc, and PhD from Imperial College, U.K., and is a Chartered Electrical Engineer. He has received 5 US patents.



Alain Tritschler is currently with the Envivo Corporation in Rennes, France. He received his MS degree in Multimedia Communications in 1998, with emphasis on speech recognition, image analysis, and networking, jointly with the Institut Eurocom, Sophia-Antipolis, France, and the Institut National des Telecommunications, Evry, France. Until 1999 he was with the Human Language Technologies department at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York. His current interests include speaker segmentation and clustering, speech recognition under noisy acoustic conditions, and multimedia retrieval.



Homayoon S.M. Beigi received his B.S., M.S. and D.Eng.Sc. from Columbia University in 1984, 1985, and 1990 respectively. He was at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York, until 1999 and is currently Chief Technology Officer at Internet Server Connections, Inc., White Plains, New York. He is also an Adjunct Associate Professor in both the EE and ME departments at Columbia University, New York. Beigi has published in the fields of Kinematics, Neural Networks, Optimization, Learning Control, Signal Processing, Image Compression, Handwriting, Speech and Speaker Recognition. He is Associate Editor of the Intelligent Automation and Soft Computing and is on the technical boards of three major international conferences.