Multimed Tools Appl (2010) 48:123–140
DOI 10.1007/s11042-009-0377-6

Content-based search in multilingual audiovisual documents using the International Phonetic Alphabet

Georges Quénot · Tien Ping Tan · Viet Bac Le · Stéphane Ayache · Laurent Besacier · Philippe Mulhem

Published online: 10 October 2009

Abstract We present in this paper an approach based on the use of the International Phonetic Alphabet (IPA) for content-based indexing and retrieval of multilingual audiovisual documents. The approach works even if the languages of the document are unknown. It has been validated in the context of the "Star Challenge" search engine competition organized by the Agency for Science, Technology and Research (A*STAR) of Singapore. Our approach includes the building of an IPA-based multilingual acoustic model and a dynamic programming based method for searching document segments by "IPA string spotting". Dynamic programming allows for retrieving the query string in the document string even with a significant transcription error rate at the phone level. The methods that we developed ranked us as first and third on the monolingual (English) search task, as fifth on the multilingual search task and as first on the multimodal (audio and image) search task.

Keywords Audio retrieval · Multilingual · International Phonetic Alphabet · Dynamic programming · Star Challenge

1 Introduction

Audiovisual databases quite often contain documents in several languages. This is the case for instance for Internet streaming archives. It often happens that the

G. Quénot (✉) · T. P. Tan · L. Besacier · P. Mulhem
Laboratoire d'Informatique de Grenoble, BP 53, 38041 Grenoble Cedex 9, France
e-mail: Georges.Quenot@imag.fr

V. B. Le
LIMSI-CNRS, BP 133, 91403 Orsay Cedex, France

S. Ayache
Laboratoire d'Informatique Fondamentale de Marseille,
163 avenue de Luminy - Case 901, 13288 Marseille Cedex 9, France

language used in a document is unknown and that a given document contains spoken utterances in different languages. This significantly complicates the content-based search within these archives. One possibility is to apply a series of language recognizers and then apply the appropriate transcribing tool but language detectors make errors and unknown languages may be encountered. Another approach, that we could qualify "low level", is to transcribe the documents into phonetic units using a subset of the International Phonetic Alphabet (IPA) regardless of the actually spoken language. Content-based search can then be done at the level of IPA strings. This approach was promoted by the Agency for Science, Technology and Research (A*STAR) of Singapore in the context of the "Star Challenge" that they organized between March and October 2008.[1] This challenge also addressed the problem of the search in video documents using the image only and using combined audio and image information.

The Star challenge was organized as a competition for multimedia search engines. It was a bit different in its spirit from classical evaluation campaigns in the field such as those organized by NIST like TRECVID [16]. It was really a competition with conditions close to real world applications, in particular concerning the timing aspects (much more constrained), and it was less oriented towards fine method or system performance measurement and comparison. The size of test collections, the number of queries, and the number of evaluated systems in the final stages are quite small. Also, participants receive information about their own performance or even only their ranking but they do not receive the corresponding information for the other participants. There is no final workshop either in which the participants describe their methods to each other.

The challenge consisted in a series of three eliminative rounds addressing respectively audio, image and multimodal content-based search. The five best ranked teams after the three rounds were invited to participate to a final competition in Singapore in "live" conditions. The search task at the audio level came in two variants: in the first one (AT1), the query was given as a phonetic string (it could be either typed as such by a user or come from a text to phone converter); in the second one (AT2), the query was given as a spoken utterance and had to be transcribed in the same way as the audio documents. We took this opportunity for developing and testing innovative approaches in content-based audio and multimodal search.

The research in cross-lingual acoustic modeling is based on the assumption that the articulatory representations of phonemes are so similar across languages that phonemes can be considered as units which are independent from the underlying language [14]. This is a very strong assumption which would not be acceptable from a purely phonetic point of view, but that may remain acceptable from an operational point of view as far as the development of automatic speech recognition technologies is concerned. Based on this assumption, we have already proposed some methods [8] for estimating the similarities of some phonetic units (phoneme, polyphone, clustered polyphone) which can be further used in cross-lingual context dependent acoustic modeling. The general idea is that for a new language, a source/target acoustic-phonetic unit mapping table can be constructed with these similarity measures. Then, acoustic models in the target language are duplicated from the nearest acoustic

[1] http://hlt.i2r.a-star.edu.sg/starchallenge

models in the source language and optionally adapted with limited data to the target language. For this, both data-driven and knowledge-based methods can be applied to automatically or manually obtain unit similarities across languages. Such phone mapping information is also useful for non native ASR [18]. In parallel, other works, like the ones from NUS team in Singapore developed the concept of "bag of sounds", inspired from information retrieval domain, to model the specificities of a language. This concept has been shown to be very efficient in language identification for instance [9].

We describe in this paper the methods that we developed for our participation to the "Star Challenge" and how we tested them in the context of this challenge. The paper is organized as follows: in Section 2, we describe how we built multilingual acoustic models; in Sections 3 and 4, we describe approach that we used for the IPA-based search, the first one using dynamic programming and the second one using the vector space model; in Sections 5 and 6, we describe approach that we used for the visual and multimodal search; in Section 7, we describe the experiments that we carried out in the context of the Star Challenge and we present the obtained results.

2 Audio processing

2.1 General approach to deal with multilingual documents

Since the languages in the audio track were not known in advance, we decided to work on a multilingual approach for transcribing the audio track. One idea could have been to run in parallel several automatic speech recognition (ASR) systems in different languages but this was not realistic in the Star Challenge context where time and computation constraints were very challenging.

Thus, we proposed a "lower level" approach where a multilingual phonetic recognizer (with phone n-grams used as language model) was applied both on database and the queries. This recognizer has the advantages of being in principle, language independent and very fast. In practice, the models were built using inputs from a small number of languages that cover the languages expected in the context of the Star Challenge. As far as we identify, the corpora finally included English and Asian languages, such as Chinese, Malay and Hindi.

2.2 Preprocessing

First, for each video, the audio track is segmented into short segments using the following steps :

1. Audio segmentation (or Speaker Change Detection): we find points in the audio stream which are candidates for audio change or speaker change points. To do this, a distance is computed between two Gaussian modeling data of two adjacent given-length windows. By sliding both windows on the whole audio stream, a distance curve is obtained. A peak in this curve is thus considered as a speaker change point. In this step, a GLR (Generalized Likelihood Ratio) is used as a distance between two Gaussian models.
2. Segment recombination: too many speaker turn points detected during the previous step may result in a lot of false speaker change points. A segment

recombination using BIC (Bayesian Information Criterion) is needed to recombine adjacent segments uttered by the same speaker. The BIC threshold value must be tuned on a development corpus in order to reduce the number of false alarms without increasing the number of new missed detection errors. See for instance [12] for more details on this step.

3. Speaker clustering: in this step, speech segments of the same speaker are hierarchically clustered using the same BIC-based distance.
4. Viterbi re-segmentation: the previous speaker clustering step provides enough data for speakers to estimate multi-Gaussian speaker models. These models are then used during a Viterbi decoding which refines the boundaries between speakers.

Finally, the resulting audio segments roughly correspond to speaker turns and have an average duration of about 10 seconds. ASR is then applied on the audio segments resulting from this initial stage. No music detection and removal is applied on the audio track.

2.3 Automatic speech recognition (ASR)

For ASR, Sphinx speech recognition system[2] with Sphinx-3 decoder[3] from Carnegie Mellon University (CMU) was selected for transcribing automatically the audio documents (database) and queries in the 1st knockout round (monolingual voice search tasks) and during the "qualifying race" (multilingual voice/video search tasks). In fact, Sphinx-3 is a fast speech decoder, capable of decoding and transcribing speech documents in real time. This is achieved using conventional Viterbi search strategy and beam heuristics. In addition, it has a lexicon-tree search structure. Sphinx-3 uses the acoustic models created by SphinxTrain and accepts n-gram language models in binary format, which are converted from a standard ARPA n-gram model.

The front-end module is used to preprocess the raw speech at 16 bits sample with sampling frequency of 16 kHz to Mel-Frequency Cepstral (MFC) feature vectors together with their first and second derivatives. This produces feature vectors with a total of 39 dimensions. SphinxTrain makes use of the feature vectors to create a continuous HMM acoustic model. Phones were used as the unit of HMM, each having three states with a left-to-right topology. Conversely, the n-gram language model (made of phone units in our case) was created using CMU statistical language modeling toolkit [3] and the SRILM toolkit [17].

2.4 Monolingual task

For the monolingual (native and dialectal English) voice search tasks, the native English acoustic models were composed of 4000 tied-states, each consisting of a mixture of 16 diagonal-covariance Gaussian densities. They have been trained using 140 hours of 1996 and 1997 HUB-4 broadcast news training data [6] by Carnegie Melon University [13]. We adapted these native English models with a small set of dialectal

[2] http://www.speech.cs.cmu.edu/sphinx/

[3] http://cmusphinx.sourceforge.net/sphinx3/doc/s3_overview.html

Table 1 Corpora used for the training of the IPA transcriber

Corpora	Description	Spk.	Hours
HUB4	English, broadcast news	–	140
WSJ0	English, read speech	123	15
VN	Vietnamese	29	15
CADCC	Chinese	20	5
MSC	Malay	18	5

English (South East Asia region) speech data by a supervised MAP adaptation. The HUB-4 language models[4] and the CMU (Carnegie Mellon University) Pronouncing Dictionary[5] which contains over 125,000 words were used.

2.5 Multilingual task

For the multilingual voice/video search tasks, since no information about the input languages was given a priori, we decided to build multilingual models from four languages: English, Mandarin, Vietnamese and Malay (corresponding, we hope, to a large coverage of what can be found in the Singapore area). The multilingual acoustic models are all context independent models trained independently with 16 Gaussians mixtures, except for English acoustic models. Two English acoustic models were actually used, namely the HUB-4 context dependent acoustic model with 8 Gaussian mixtures from CMU, which was trained from broadcast news, and a context independent acoustic model with 8 Gaussian mixtures, which was trained from WSJ0 read corpus [5]. Since the HUB4 acoustic model from CMU is a context dependent model, we extracted only the context independent states out. The Mandarin acoustic model was trained from CADCC corpus [2], the Vietnamese acoustic model was trained from VnSpeechCorpus [7] and the Malay acoustic model was trained from a Malay corpus courtesy of University Sains Malaysia. Table 1 shows some characteristics of the used corpora including the number of speakers and the number of hours of audio signal. Finally, the total number of phone units represented is 147 (39 English units, 40 Vietnamese units, 35 Malaysian units and 33 Chinese units).

For language modeling, a multilingual phone-based bigram model was trained from multilingual text corpora for 4 languages. The use of a phone-based language model speeds up very significantly the speech decoder (around 0.25 RT).

3 Dynamic programming based search

The search is always done at the level of IPA strings regardless of whether the transcription was initially done at the word level or at the phone level and regardless of whether the query was made as an IPA string (AT1) or as a spoken utterance (AT2). In all cases, we have to match and score IPA representations of both queries

[4] http://www.speech.cs.cmu.edu/sphinx/models/

[5] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

and documents. We have chosen to do it by using a variant of a word spotting algorithm [4]. The main difference between the original word spotting algorithm and our "IPA string spotting" algorithm is that we replace the audio feature vectors (typically Mel Frequency Cepstral Coefficients or MFCCs) by the IPA symbols.

3.1 Minimization of the edit distance

Due to frequent transcription errors either in the documents for both tasks and/or in the queries for AT2, the search for the query string within the document strings must allow inexact matches. Matches, either exact or inexact, need also to be scored so that the documents with the most exact matches can be ranked first. In order to allow for inexact match and to score them, we chose the "edit distance" (also called Levenstein distance) between a query string and a substring of a document string. All the possible matches between a query string and all existing substrings from all the document strings are considered during search. For each match, a distance is computed by counting and penalizing all the insertions, deletions and substitutions between the phone units composing the query string and the document substring. Figure 1 shows an example of edit distance computation (note that the query is logically considered as the reference string during the edit distance computation).

3.2 Dynamic programming

Dynamic programming is a way to solve the problem of finding and scoring the best alignment between a query string and a document substring in a computing time that is linear in both the query string length and the document string length.

The product matrix between the document string (horizontal) and the query string (vertical) is considered. A valid matching or alignment between the query string and a substring of the document string is an increasing path that joins the matrix bottom row to the matrix top row (Fig. 2). The best alignment is the one that minimizes the edit distance along itself. The dynamic programming trick is to compute the best alignment by recurrence.

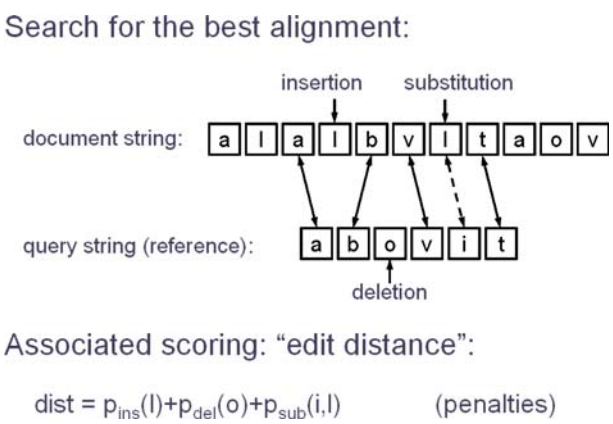

Fig. 1 Scoring of a matching between a query string and a substring of a document string

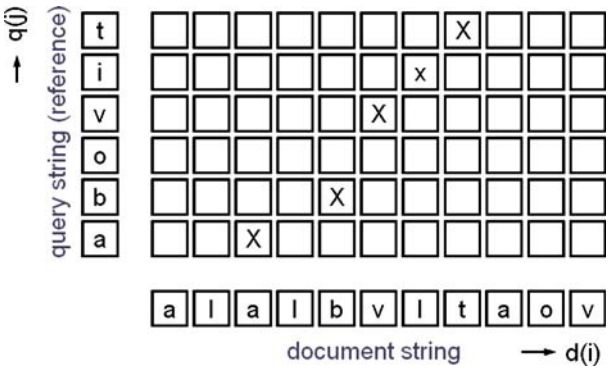

Fig. 2 Matching as a path in the DP product matrix

If we consider the best edit distance $e(i, j)$ from the bottom row to the (i, j)-point, we have a recurrence equation on $e(i, j)$ since the best path arriving at the (i, j)-point must either:

- come from $(i - 2, j - 1)$ with an insertion penalty,
- come from $(i - 1, j - 2)$ with a deletion penalty or
- come from $(i - 1, j - 1)$ with a possible substitution penalty.

(unless one of these points is outside of the matrix).

$e(i, j)$ can be computed by recurrence in the whole matrix with an initialization to 0 on the bottom row and to "infinite" on the left column (excluding the bottom value). The actually used recurrence equation is given in Eq. 1. The c_{xx} are constants whose values are: $c_{ii} = c_{dd} = 2.0$, $c_{sn} = c_{sd} = 1.0$, $c_{si} = 0.5$ (normalization according to the query so that all alignments have the same total weight and the same weight for the insertion, deletion and substitution penalties).

$$e(i, j) = \min \left\{ \begin{array}{l} e(i-2, j-1) + c_{si}(p_{sub}(d(i-2), q(i-1)) \\ \qquad\qquad + p_{sub}(d(i), q(i)) + c_{ii}\, p_{ins}(d(i-1))) \\ e(i-1, j-1) + c_{sn}(p_{sub}(d(i-1), q(i-1)) \\ \qquad\qquad + p_{sub}(d(i), q(i))) \\ e(i-1, j-2) + c_{sd}(p_{sub}(d(i-1), q(i-2)) \\ \qquad\qquad + p_{sub}(d(i), q(i)) + c_{dd}\, p_{del}(q(i-1))) \end{array} \right\} \quad (1)$$

Once done, the minimum of $e(i, j)$ on the top row gives the best edit distance and query score (the lower, the better). Backtracking from the minimum location gives the location of the instance of the best match.

3.3 Variable versus fixed penalties

The phone insertion, deletion and substitution penalties may be either constant or may depend upon the actually inserted, deleted or substituted phones since some phones are more likely than other to be inserted, deleted or substituted. For the fixed penalties, we chose:

- $p_{ins(p_i)} = 1$
- $p_{sub(p_i, p_j)} = 1 - \delta(i, j)$
- $p_{del(p_j)} = 1$

and for the variable penalties, we chose:

- $p_{ins(p_i)} = -\log(\epsilon + prob\,(insertion(p_i))$
- $p_{sub\,(p_i,p_j)} = -\log(\epsilon + prob\,(substitution(p_i, p_j)))$
- $p_{del(p_j)} = -\log(\epsilon + prob\,(deletion(p_j))$

with $\delta(i, j) = 1$ if $i = j$ and 0 otherwise. The probabilities were derived from the error rate between manual (exact) and automatic transcriptions.

4 Vector space model based search

We also made experiments with the Vector Space Model (VSM) classically used in textual information retrieval. We used phone bigrams as the basic indexing unit associated to a pivoted cosine normalization scheme. This normalization was proposed by Singhal, Buckley and Mitra in 1996 [15]. In the vector space model, we usually define, for the weighting scheme, a normalization factor intended to compensate for the fact that long documents have intrinsically more chances to be relevant for a given query. Such normalization may use the document vector norm or the size of the document in terms of characters for instance. In [15], the authors found out that using usual cosine normalization tends to over boost the short documents and to penalize too much the long documents. This is why they proposed a pivoted normalization to compensate for the usual normalization. In our case, documents vary a lot in length, explaining why such pivoted length normalization is needed. Compared to usual normalization slope for text (between 0.2 and 0.3 according to [15]) the slopes of the normalization for our best results using the vector space model are of 0.4 for AT1 and 0.54 for AT2. This normalization slope is quite large; this is probably due to the fact that, on this collection, longer documents need to be boosted during the indexing to be retrieved.

5 Image and video content-based search

The system performs visual concept classification by supervised learning using standard annotation provided by the Star Challenge organizers. Our generic classification system is defined by a network of operators [1], which includes low-level features extractor, mid-level semantic classification and fusion classifiers. The following subsections describe those three steps.

5.1 Visual analysis

We performed visual analysis at several levels of granularity from global to fine blocs analysis, as well as various semantic levels. Our low-level feature extractors first split images on overlapped blocs to form a grid of N×M blocs. For our submissions, we chose N and M such as we obtained a satisfying trade-off between classification performance and time computing. Finally, those values depends on which feature is considered. During the analysis, firstly each key frame is processed to extract several feature vectors; secondly, these features are merged using standard early or late fusion schemes, or a combination of them; finally, key frames are merged and a score is assigned to each shot.

5.1.1 Low-level features

At global level, we consider classical color and texture features. Color is represented by a 3-dimensional histogram on RGB space. We discretize the color space to form a 4×4×4 bins histogram. Texture information is described with Gabor bank of filters; we used 8 orientations and 5 scales. Finally, global features are normalized and concatenated on a 104 dimensions vector.

We also extracted color and texture features at block levels, features obtained from each block are then concatenated to form a rich description of key frames:

Color (1):	is represented by 3×3×3 3D histogram on a grid of 8×6 blocs. The overall local color feature vector has 1296 dimensions.
Color (2):	is represented by the two first moments on a grid of 8×6 blocs. This local color feature vector has 432 dimensions.
Edge Direction Histogram:	is computed on a grid of 4×3 blocs. Each bin is defined as the sum of the magnitude gradients from 50 orientations. Thus, overall EDH feature has 600 dimensions. EDH feature is known to be invariant to scale and translation.
Local Binary Pattern:	Mäenpää Topi Pietikäinen Matti [11] is computed on grid of 2×2 blocs, leading to a 1024 dimensional vector. The LBP operator labels the pixels of an image by thresholding the 3×3-neighborhood of each pixel with the center value and considering the result as a decimal number. LBP is known to be invariant to any monotonic change in gray level.

5.1.2 Feature on interest points

One of the more relevant feature for visual indexing is the SIFT descriptor combined with a "bag of words" representation. The SIFT descriptor [10] describes the local shape of points of interest using edge histograms. To make the descriptor invariant, the interest region is divided into a 4×4 grid and every sector has its own edge direction histogram (8-bin). We used a codebook of 1000 visual word.

5.2 Semantic feature

This feature aims at modeling co-occurrence between high-level features using a "bag of concepts" approach. First, we consider each block from key frames which are relevant for a concept, as relevant for this concept too. This is a very strong assumption but it could be reasonable depending of the concepts. Thus, we use existing concepts annotations (from a part of the learning set) at global level, to train SVM classifiers at the blocks level, where blocs are represented with moments color and edge direction histogram features. Then blocs of key frames are classified using models of all the concepts, leading to $nb\_blocs \times nb\_concepts$ classification scores

per key frame. The final semantic feature is defined by the sum of scores on *nb_blocs* for each concepts, leading to a *nb_concepts* dimensional feature.

5.3 Early and late fusion

We merged our various features with combinations of early and late fusion schemes. While the early fusion proceeds in the feature space, the late fusion combines classification scores. Combination of those two schemes is possible when more than two features is available and yields much flexibility on the way to merge features. For example we can combine features with early fusion once then combine with others feature with late fusion. Such combinations outperform, for some concepts, classical early and late fusion.

5.4 Optimization

With various combination of early and late fusion, we obtained numerous scores for concept detection. This process aims at selecting the best combination for each concept. We classified each shot on a subpart of the development set and selected the combination with higher performance. Those selected combinations have been used for the final classification on test set.

6 Multimodal content-based search

Video segments can be scored and sorted according to:

- the propability of presence of a given concept;
- the visual similarity to a query image or video segment;
- the propability of presence of a given phonetic string.

A mono or multimodal query can be defined as a combination of such criteria. For example, in the star challenge video tasks 1 and 2 (VT1 and VT2), a query is defined as a combination of a required visual concept and a visual similarity to a given image (VT1) or a video shot (VT2). In the multimodal search task 1 and 2 (AV1 and AV2), a query is defined as a combination of a visual similarity to a given image and the presence of a textual phonetic string (AV1) or of a spoken phonetic string (AV2).

A similar approach is used in all cases. A numerical score is obtained for each criterion using the appropriate subsystem: phonetic string search, spoken utterance search, concept indexing or visual similarity to example. These scores are normalized by criterion using a simple gain and offset normalization. A weighted sum of these normalized scores is computed and video segments are ranked according to it. The appropriate weights are determined as those maximizing the search performance on the training set.

Visual similarity is based on a Euclidian distance on color, texture and motion descriptors used for concept classification. Visual similarity is computed separately for each component and the corresponding scores are normalized and combined in the same way as for multimodal components. Again, the optimal weights for the sum are determined as those maximizing the search performance on the training set.

7 Experimentations

7.1 Monolingual search, validation on the Star Challenge development set

The goal of the first series of experiments was to evaluate the relative performance of word-based and phone-based search as well as the benefit brought by the use of variable penalties in the latter case. Three dynamic programming (DP) based methods were compared to some baselines and to a Vector Space Model based methods.

These experiments were carried out on the audio development set of the Star Challenge. This set consists in about two hours of monolingual (English) audio data (233 segments) and 39 solved queries for both AT1 (queries as IPA strings) and AT2 (spoken queries). The systems are asked to return a list on 50 hits and the evaluation metric is the MAP defined in the Star Challenge for audio search (Eq. 2; this MAP is different from the standard TREC MAP metric):

$$MAP = \frac{1}{L}\sum_{i=1}^{L}\left(\frac{1}{R_i}\sum_{j=1}^{R_i}\delta(i,\,j)\right) \tag{2}$$

where L denotes the total number of queries, R_i is the total number of documents relevant to the ith query, and $\delta(i,\,j)$ is an indicator function which is 1 when there is a hit (i.e. the jth relevant document is in the list output by the retrieval method for query i) and 0 otherwise.

Documents (audio segments) and AT2 queries were transcribed in IPA in two ways. The first one (slower) is a transcription at the word level followed by a conversion at the phone level. The second one (faster) is a direct transcription at the phone level. Thus, the difference between the first and the second approaches is the use (or not) of a word language model during ASR. After that and in both cases, all documents and queries are constituted of IPA strings.

Several baselines were used for comparison. A random choice corresponds to baseline 2. Another possibility is to sort the segments according to their length: the longer the segment, the higher its probability to contain the query, regardless of its contents. Baseline 1 corresponds to the choice of the shortest segments (worst case) and baseline 3 corresponds to the choice of the longest segments (best case). All of these baselines actually ignore the contents of the segments and the results are the same for AT1 and AT2 since the make no use of the query at all. Baseline 4 corresponds to the search for an exact match of the query string within the segment string. This corresponds to the Unix "grep" command and also to a dynamic programming based search with infinite deletion, suppression and substitution penalties. Since exact matches are very infrequent and do not provide enough segments for completely filling the result list, additional segments are chosen among the longest ones.

Dynamic programming was tried at the word level with fixed and variable penalties and at the phone level with variable penalties.

Finally, the Vector Space Model (VSM) commonly used in information retrieval was tried. The indexing terms were bigrams of phones with the pivoted cosine normalization weighting scheme.

Table 2 Validation on the Star Challenge development set

Method	AT1	AT2
Baseline 1: shortest segments	0.024	0.024
Baseline 2: random	0.242	0.242
Baseline 3: longest segments	0.497	0.497
Baseline 4: "grep" + longest segments	0.557	0.560
DP, word recog., fixed penalties	0.776	0.632
DP, word recog., variable penalties	0.843	0.636
DP, phone recog., fixed penalties	0.643	0.633
DP, phone recog., variable penalties	0.706	0.650
VSM, word recog., bigrams	0.797	0.660
VSM, phone recog., bigrams	0.552	0.543

Table 2 shows the results obtained for the tested methods and baselines. The following observations can be made:

- the baseline performance increased as we predicted: shortest < random < longest < grep+longest;
- the variable penalties significantly improved the performance;
- as expected too, the phone level transcription produces lower performance results than the word level one, because it does not benefit from the word language model during ASR; however, it is the only method that is realistic for documents containing unknown languages and it will be used for the multilingual search;
- finally, the VSM based systems performed quite well at the word level but it did less well than baseline 4 at the phone level, probably due to the usually better quality of the language mode at word level.

7.2 Monolingual search, evaluation on the Star Challenge "round 1" data set

The system that performed better on the development set was used for the official submission of the round 1 of the Star Challenge. This is a system using dynamic programming for the search with variable penalties. An additional improvement was made; it consisted in using multiple (three) transcriptions with different phone bigram weights and averaging over the corresponding DP scores. This leaded to another slight improvement in the system performance.

Table 3 shows the results obtained with this system on the round 1 set. This set is composed of 25 hours of monolingual (English) audio data (4300) segments and 10 queries for both AT1 and AT2 tasks. For comparisons, results are also shown with

Table 3 Influence of the use of variable penalties

Data set	Penalties	AT1	AT2	Mean
Devel.	Fixed	0.760	0.679	0.719
Devel.	Variable	0.858	0.728	0.793
Round 1	Fixed	0.643	0.319	0.481
Round 1	Variable	0.634	0.324	0.479

the same system with fixed penalties on round 1 data and with fixed and variable penalties on development data. The following observations can be made:

- the performance on round 1 data is much lower than on development data;
- the significant performance gain obtained by the use of variable penalties on development data is not obtained again on round 1 data;
- the performance drop between AT1 and AT2 is much more important on round 1 data.

All these effects are probably due to the fact that round 1 data contains much more audio segments of which a much smaller proportion is relevant, making the task harder. These segments are also smaller. Using this approach, the LIG team ranked first on AT1, third on AT2 and third globally for a total of 35 participating teams.

7.3 Multilingual search, validation on the Star Challenge "round 1" data set

The goal of this series of experiments is to validate the multilingual search using the available training data. Since the target languages were not known we could only validate the approach on the monolingual (English) available data. We did however build models using other languages and tested it on English data, considering that this would be representative enough. We first tried models built from a single language and used them for indexing and retrieval: English (EN), Chinese (CH) and Malay (MY). We also tried a model which is a combination of these three languages and Vietnamese (ML4). We finally tried the same with phone bigrams (BG) and a combination of three models (EN, CH and MY) with different language model weighting (Fuse).

Table 4 shows the results obtained with these different models. The following observation can be made:

- we used a new English model that turned out to be better than the one used for the official submission on round 1, especially for AT2;
- the results obtained with phone decoders trained on languages that are different from the target language (Chinese or Malay against English) are quite good despite the very different phonetic contents;
- results are better for AT2 than for AT1 in these cases (CH and MY); this is probably due to the fact that similar confusions are made during the document and query transcription and that they compensate each other;
- the ML4 multilingual model is almost as good as the purely English model and BG and Fuse perform even better; based on this result, these models are expected to be also acceptable for Asian languages.

The LIG team used the "Fuse" model as its official submission for the round 3 and ranked fifth on the audio task and first on the multimodal task (combination of IPA based search and image content-based search of video segments), qualifying for the challenge final in Singapore.

Table 4 Results obtained with different language models

	EN	CH	MY	ML4	BG	Fuse
AT1	0.668	0.476	0.428	0.603	0.615	0.650
AT2	0.585	0.578	0.577	0.568	0.591	0.638

7.4 Visual search, evaluation on the Star Challenge "round 2" data set

The visual search task VT1 was to find images (actually video shot key frames) that contained a given concept (within 20 for which the system was trained) and that were visually similar to a given image. We found that the best results were obtained by weighting 2:1 the normalized visual similarity and concept presence probability.

The visual search task VT2 was to find video shots that contained a given concept (within 10 for which the system was trained) and that were visually similar to a given video shot. We found that the best results were again obtained by weighting 2:1 the normalized visual similarity and concept presence probability.

This approach ranked us fifth in both VT1 and VT2 tasks.

7.5 Multimodal search, evaluation on the Star Challenge "round 3" data set

The multimodal search task AV1 (resp. AV2) was to find video shots that were visually similar to a given image and that contained a given audio query defined as a textual IPA string (resp. a spoken utterance). We found that the best results were obtained by weighting 30:70 the normalized visual similarity and IPA spotting detection scores.

This approach ranked us first on the multimodal task (combined AV1 and AV2).

8 Conclusion

We have presented an approach based on the use of the International Phonetic Alphabet (IPA) for content-based indexing and retrieval of multilingual audiovisual documents. The approach works even if the languages of the document are unknown. It has been validated in the context of the "Star Challenge" search engine competition organized by the Agency for Science, Technology and Research (A*STAR) of Singapore. Our approach includes the building of an IPA-based multilingual acoustic model and a dynamic programming based method for searching document segments by "IPA string spotting". Dynamic programming allows for retrieving the query string in the document string even with a significant transcription error rate at the phone level. The methods that we developed ranked us as first and third on the monolingual (English) search task, as fifth on the multilingual search task and as first on the multimodal (audio and image) search task. Both the audio and image indexing processes run in about the same time as the video collection total duration as this was the constraint during the Grand Final of the Challenge.

The results obtained were quite good and validated the principle of the use of IPA transcriptions for the indexing and retrieval of audiovisual documents in unknown languages. Some additional experiments should be done on larger corpora for confirming the obtained results. Several improvements are still possible both at the level of the building of the multilingual acoustic models and at the level of the dynamic programming based search. On the latter point, a direct search in the phone lattice produced by the transcribing tools appears very promising.

The combination of IPA spotting, concept indexing and visual similarity proved to be very efficient for multimodal content based search in video documents.

Acknowledgement Part of this work has been supported by the Quaero programme.

References

1. Ayache S, Quénot G (2007) Image and video indexing using networks of operators. J Image Video Process 2007(4):1–13. doi:10.1155/2007/56928
2. CCC (2005) http://www.dear.com/CCC/resources.htm
3. Clarkson P, Rosenfeld R (1997) Statistical language modeling using the CMU-Cambridge toolkit. In: Eurospeech'07, pp 2707–2710
4. Gauvain JL, Mariani JJ (1982) A method for connected word recognition and word spotting on a microprocessor. In: Proc. IEEE ICASSP 82, vol 2, pp 891–894
5. LDC (1993) http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S6B
6. LDC (1997) http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC98S71
7. Le VB, Do-Dat T, Casteli E, Besacier L, Serignat JF (2004) Spoken and written language resources for Vietnamese. In: LREC'04, pp 599–602
8. Le VB, Besacier L, Schultz T (2006) Acoustic-phonetic similarities for context dependent acoustic model portability. In: Proc. IEEE ICASSP 2006
9. Li H, Ma B, Lee CH (2007) A vector space modeling approach to spoken language identification. IEEE Transactions on Audio, Speech and Language Processing 15:91–110
10. Lowe D (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60:91–110
11. Mäenpää Topi Pietikäinen Matti OT (2000) Texture classification by multi-predicate local binary pattern operators. In: 15th international conference on pattern recognition, vol 3, pp 951–954
12. Moraru D, Besacier L, Meignier S, Fredouille C, Bonastre JF (2004) Speaker diarization in the ELISA consortium over the last 4 years. In: RT2004 fall workshop
13. Placeway P, Chen S, Eskenazi M, Jain U, Parikh V, Raj B, Ravishankar M, Rosenfeld R, Seymore K, Siegler M, Stern R, Thayer (1997) The 1996 hub-4 sphinx-3 system. In: In DARPA speech recognition workshop. Chantilly
14. Schultz T, Waibel A (2001) Language independent and language adaptive acoustic modeling for speech recognition. Speech Commun 35:31–51
15. Singhal A, Buckley C, Mitra A (1996) Pivoted document length normalization. In: ACM SIGIR conference. ACM, New York, pp 21–29
16. Smeaton AF, Over P, Kraaij W (2006) Evaluation campaigns and trecvid. In: MIR'06: proceedings of the 8th ACM international workshop on multimedia information retrieval. ACM, New York, pp 321–330. doi:10.1145/1178677.1178722
17. Stolcke A (2002) SRILM—an extensible language modeling toolkit. In: Intl. conf. on spoken language processing. citeseer.ist.psu.edu/stolcke02srilm.html
18. Tan TP, Besacier L (2008) Improving pronunciation modeling for non-native speech recognition. In: Interspeech 2008

Georges Quénot is Researcher at CNRS (French National Centre for Scientific Research). He has an engineer diploma of the French Polytechnic School (1983) and a PhD in computer science (1988) from the University of Orsay. He is currently with the Multimedia Information Indexing

and Retrieval group (MRIM) of the Laboratoire d'informatique de Grenoble (LIG) where he is responsible for their activities on video indexing and retrieval. His current research activity is about semantic indexing of image and video documents using supervised learning, networks of classifiers and multimodal fusion.

Tien Ping Tan is currently a researcher and lecturer in Universiti Sains Malaysia, Malaysia. He obtained his PhD in computer science from Joseph Fourier University, Grenoble, France in 2008. His research interest is in the field of multilingual automatic speech recognition, speech search and Malay speech processing.

Viet Bac Le received his Master and Ph.D. degree in Computer Science from the Joseph Fourier University, Grenoble, France, in 2002 and 2006. His Ph.D. thesis was "Automatic Speech Recognition for Under-Resourced Languages". From 2006 to 2008, he was post-doc fellow at the Lorraine Laboratory of IT Research and its Applications (LORIA), Nancy, France and at the Grenoble Informatics Laboratory (LIG), Grenoble, France. He is now with the Computer Sciences Laboratory for Mechanics and Engineering Sciences (LIMSI), Orsay, France. His research interests include Speech Transcription, Speaker Diarization, Identification and Tracking for Spoken Documents.

Stéphane Ayache received a Master in Computer Science in 2003 from the University of Joseph Fourier (Grenoble, France). In 2007, he received a PhD degree from the Grenoble Institute of Technology (INPG, France) on "Semantic video indexing by combination of Image, Audio and Text features". He then was post-doc at Lyon Research Center for Images and Intelligent Information System (LIRIS). From 2008, he is an Assistant Professor at Aix-Marseille University and is member of the Fundamental Computer Science Lab (LIF) in Marseille, France. His research interests include semantic multimedia indexing, multimedia retrieval and machine learning models.

Laurent Besacier received his Ph.D. degree in Computer Science in 1998 from the University of Avignon (France) on "A parallel model for automatic speaker recognition". Then he spent 18 months at IMT (Switzerland) as an Associate Researcher. Since 1999, he is an Associate Professor in Computer Science at the University Joseph Fourier (Grenoble, France). From 2005 to 2006, he was Invited Scientist at IBM Watson Research Center working on Speech to Speech Translation. He defended his "Habilitation à Diriger les Recherches" (HDR) in 2007 on "Rich Transcription in a Multimodal and Multilingual World". His research interests can be divided in two main parts: speech and audio processing in a multimodal framework, as well as multilingual speech recognition and (recently) translation.

He has published around 100 papers in the best journals (Speech Communication, Computer Speech and Language) and conferences (12 ICASSP papers, 14 Interspeech papers) of the domain and supervised or co-supervised 9 PhD students and 9 Master students.

He has also been involved in several national and international projects : among others, one can quote NESPOLE European project on speech-to-speech translation, M2VTS European project on multimodal biometrics, as well as evaluation campaigns organized by NIST or DARPA : RT03, RT04, RT05, TRECvid, TRANSTAC.

Philippe Mulhem is currently researcher of the French National Center for Scientific Research at the Computer Science Laboratory of Grenoble (LIG) in the Modeling and Multimedia Information Retrieval group. He was formerly, during 5 years, director of the Image Processing and Applications Laboratory in Singapore, a joint laboratory between the French National Center of Scientific Research, the National University of Singapore, and the Institute for Infocomm Research of Singapore. His research interests include formalization and experimentation of image, video, and structured multimedia document indexing and retrieval. He is involved in several international and national projects related to these fields. He is author or co-author of more than 80 papers in international and national journals, conference proceedings and book chapters.