Recherche par le contenu dans des documents audiovisuels multilingues

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

** Laboratoire d'Informatique de Grenoble*
BP 53, 38041 Grenoble cedex 9

Georges.Quenot imag.fr

*** Laboratoire d'Informatique pour la Mécanique*
et les Sciences de l'Ingénieur
BP 133, Bât 508 F-91403 Orsay cedex

**** Laboratoire d'Informatique Fondamentale de Marseille*
163 avenue de Luminy - Case 901, F-13288 Marseille cedex 9

RÉSUMÉ. *Nous présentons dans cet article une approche basée sur l'utilisation de l'alphabet phonétique international (API) pour l'indexation et la recherche par le contenu de documents audiovisuels multilingues. Elle a été validée lors de la compétition « Star Challenge » sur les moteurs de recherche organisée par l'Agence A*STAR de Singapour. Elle comprend la construction d'un modèle acoustique multilingue basé sur l'API et une méthode basée sur la programmation dynamique pour la recherche de segments de documents par « détection de chaînes API ». Les méthodes que nous avons développées ont obtenu de très bons résultats sur l'ensemble des tâches du challenge.*

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. It has been validated within the "Star Challenge" search engine competition organized by the A*STAR Agency of Singapore. It includes the building of an IPA-based multilingual acoustic model and a dynamic programming based method for searching document segments by "IPA string spotting". The methods that we have developed have obtained very good results on all the tasks of the challenge.*

MOTS-CLÉS : *recherche audio, multilingue, alphabet phonétique international, programmation dynamique, star challenge.*

KEYWORDS: *audio retrieval, multilingual, international phonetic alphabet, dynamic programming, star challenge.*

DOI:10.3166/DN.13.1.229-246 © 2010 Lavoisier, Paris

RSTI - DN – 13/2010. Recherche d'information, pages 229 à 246

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

1. Introduction

Les bases de données audiovisuelles contiennent souvent des documents en plusieurs langues. C'est le cas par exemple pour les archives sur internet. Il arrive souvent que la langue utilisée dans un document soit inconnue et que le document contienne des énoncés prononcés dans différentes langues. Cela complique la recherche par le contenu dans les archives. Une possibilité consiste à appliquer un système de reconnaissance de la langue pour ensuite appliquer le système de transcription approprié mais les détecteurs de langue commettent des erreurs et des langues inconnues peuvent être rencontrées. Une autre approche consiste à transcrire les documents phonétiquement en utilisant un sous-ensemble de l'alphabet phonétique international (API), quelle que soit la langue parlée. La recherche par le contenu peut alors être effectuée au niveau des chaînes de caractères en API. Cette approche a été encouragée par l'agence de la science, de la technologie et la recherche (A*STAR) de Singapour dans le cadre du défi « Star Challenge » qu'elle a organisé entre Mars et Octobre 2008[1]. Ce défi a également abordé le problème de la recherche dans des documents vidéo en utilisant uniquement l'image ou en utilisant des informations combinées de l'audio et de l'image.

Le défi « Star Challenge » est organisé comme une compétition pour les moteurs de recherche multimédias. Il est un peu différent dans l'esprit des campagnes d'évaluation classiques dans ce domaine telles que celles organisées par le NIST. Il s'agit vraiment d'une compétition dans des conditions proches de celles des applications du monde réel, en particulier en ce qui concerne les aspects temps de traitement (bien plus réduits). Elle est moins orientée vers une mesure précise de la performance des méthodes ou des systèmes. Le défi consiste en une série de trois rounds éliminatoires portant sur la recherche par le contenu dans des documents audiovisuels respectivement par l'audio, l'image et la combinaison des deux. Les cinq meilleures équipes classées après les trois rounds ont été invitées à participer à une épreuve finale « en direct » à Singapour. La tâche de recherche par l'audio existe en deux variantes : dans la première (AT1), la requête est fournie sous la forme d'une chaîne de caractères phonétiques (qui pourrait être entrée telle quelle par un utilisateur ou provenir d'une conversion à partir du texte) ; dans la seconde (AT2), la requête est fournie sous la forme d'un énoncé audio et doit être transcrite de la même manière que les documents audio. Nous avons profité de cette occasion pour développer et tester des approches innovantes pour la recherche par le contenu audio et multimodal dans des bases de recherche audio.

Nous décrivons dans cet article les méthodes que nous avons développées pour cette participation et la façon dont nous les avons testées dans le cadre de ce défi. L'article est organisé comme suit : dans la section 2, nous décrivons comment nous avons construit nos modèles acoustiques multilingues ; dans les sections 3 et 4, nous décrivons deux approches que nous avons utilisées pour la recherche par API, la première basée sur la programmation dynamique et la seconde basée sur le modèle vectoriel de

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

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

recherche d'information ; dans les sections 5 et 6, nous décrivons les approches que nous avons utilisées pour la recherche visuelle et multimodale ; dans la section 7, nous décrivons les expériences que nous avons effectuées dans le cadre du Star Challenge et nous présentons les résultats obtenus.

2. Traitement de l'audio

2.1. *Traitement des documents multilingues – approche générale*

Comme les langues parlées dans les documents audio sont supposées inconnues au départ, nous avons envisagé une approche multilingue pour la transcription automatique de documents audio. En effet, une solution aurait consisté à utiliser en parallèle différents systèmes de reconnaissance monolingue, mais celle-ci n'était pas réaliste dans le contexte de la compétition Star Challenge où le temps de calcul était une contrainte très importante.

Par conséquent, nous avons envisagé une approche plus « bas niveau » où un décodeur phonétique multilingue a été utilisé pour transcrire les documents et les requêtes. Ce décodeur a l'avantage d'être en principe indépendant de la langue et très rapide. En réalité, ce décodeur de phonème n'est pas tout à fait indépendant de la langue car il dépend d'un ensemble de langues sources utilisées pour entraîner les modèles acoustiques et les modèles de langage « phonémiques ».

2.2. *Système de transcription automatique de la parole*

Pour chaque document audio, le signal audio a été extrait et segmenté en segments homogènes (parlés par un seul locuteur) en utilisant un système de segmentation audio fondé notamment sur le critère BIC (*Bayesian Information Criterion* - voir (Moraru *et al.*, 2004) pour plus de détails). En principe, un segment obtenu par ce système correspond à un tour de parole. Ensuite, un décodeur de parole a été appliqué sur chaque segment. Aucun détecteur de musique ou de silence n'a été utilisé ici pour enlever les segments ne contenant pas de parole.

Le décodeur Sphinx-3 [2] de Carnegie Mellon University (CMU) a été utilisé pour transcrire automatiquement des documents audio et des requêtes du round 1 (tâches de recherche vocale monolingue) et de la phase de qualification (round 3) pour la finale (tâches de recherche vocale/vidéo multilingue). En fait, le décodeur Sphinx-3 est un décodeur rapide qui fonctionne en temps réel. Il implémente une stratégie de recherche en faisceaux *via* l'algorithme de Viterbi avec un contrôle de la largeur des faisceaux (*Beam-Search*) à plusieurs niveaux (état HMM, phonème, mot ou phrase). Sphinx-3 utilise les modèles acoustiques HMM créés et entraînés par SphinxTrain et il accepte en entrée des modèles de langage n-grammes ARPA standard au format binaire.

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

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

Un module de paramétrisation du signal a été utilisé pour extraire toutes les 10 ms sur une fenêtre d'analyse un vecteur acoustique. Chaque vecteur acoustique consiste en 13 coefficients MFCCs, les dérivées première et seconde de ces coefficients pour obtenir finalement un ensemble de 39 paramètres. Toutes les unités acoustiques ont été construites sur une topologie de HMMs continus gauche-droit d'ordre 1 à 3 états où chaque état est une distribution multi-gaussienne. Pour apprendre les modèles de langage n-grammes, nous avons utilisé les boîtes à outils SRILM (Stolcke, 2002) et CMU (Clarkson *et al.*, 1997).

2.3. *Tâche monolingue*

Pour les tâches de recherche vocale (*voice search*) monolingue (anglais natif et dialectal), des modèles acoustiques anglais de 4 000 états (*tied-states*) ont été utilisés. Chaque état a été modélisé par un mélange de 16 distributions gaussiennes à matrice de covariance diagonale. Ces modèles acoustiques ont été créés par Carnegie Melon University (Placeway *et al.*, 1997) et ils ont été appris à partir du corpus d'apprentissage broadcast news HUB-4 1996-1997 (LDC, 1997) qui contient 140 heures de signal de parole. Ensuite, ces modèles natifs anglais ont été adaptés par nos soins avec la méthode d'adaptation supervisée MAP (Maximum a Priori) (Gauvain *et al.*, 1994) en utilisant une petite quantité de données de parole dialectale de la région de l'Asie du Sud-Est. Par ailleurs, le modèle de langage HUB-4 [3] et le grand dictionnaire de prononciation de CMU [4] avec 125 000 mots ont été utilisés.

2.4. *Tâche multilingue*

Pour les tâches de recherche vocale multilingue, comme les langues parlées dans les documents audio sont supposées inconnues au départ, nous avons décidé de construire des modèles acoustiques multilingues pour 4 langues : anglais, mandarin, vietnamien et malais. Nous pensions que ces 4 langues seraient largement utilisées dans les documents audiovisuels dans la région de l'Asie du Sud-Est et la région Singapourienne en particulier.

Les modèles acoustiques multilingues indépendants du contexte sont entrainés séparément pour le mandarin, le vietnamien et le malais avec un mélange de 16 distributions gaussiennes pour chaque état du modèle HMM. Le modèle acoustique mandarin a été appris à partir du corpus CADCC (CCC, 2005), le modèle vietnamien a été appris à partir du corpus VnSpeechCorpus (Le *et al.*, 2004) et le modèle malais a été appris à partir d'un corpus donné par l'Université Sains Malaysia. Un modèle acoustique anglais indépendant du contexte avec un mélange total de 16 distributions gaussiennes a été combiné à partir de deux modèles acoustiques différents : HUB-4 (issu de CMU, de type broadcast news) avec 8 gaussiennes et WSJ0 avec 8 gaussiennes (de type parole

3. http ://www.speech.cs.cmu.edu/sphinx/models/
4. http ://www.speech.cs.cmu.edu/cgi-bin/cmudict

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

lue) (LDC, 1993). Comme le modèle HUB-4 est originalement un modèle dépendant du contexte, nous n'avons extrait que les parties indépendantes du contexte à partir de ce modèle. Le tableau 1 présente le nombre de locuteurs et la taille des corpus de parole utilisés.

Enfin, un modèle acoustique multilingue est composé à partir de l'union des 4 modèles acoustiques monolingues. Les étiquettes linguistiques ont été ajoutées à chaque modèle de phonème pour que les phonèmes venant de langues différentes puissent être différenciés le cas échéant.

Corpus	Description	Nb. loc.	Heures
HUB4	Anglais, broadcast news	–	140
WSJ0	Anglais, parole lue	123	15
VN	Vietnamien, parole lue	29	15
CADCC	Mandarin	20	5
MSC	Malais	18	5

Tableau 1. *Corpus de parole utilisés pour la modélisation acoustique multilingue*

Pour la modélisation du langage, un modèle de phonème multilingue bigramme a été appris à partir du corpus de texte multilingue pour 4 langues. L'utilisation du modèle de langage phonétique a accéléré significativement le décodage de parole (le temps de calcul était environ de 0,25 $\times$ RT).

3. Recherche par programmation dynamique

La recherche est toujours effectuée au niveau des chaînes de caractères API, indépendamment du fait que la transcription ait été faite au niveau du mot ou du phonème et indépendamment du fait que la requête soit présentée comme une chaîne de caractères API (AT1) ou comme un énoncé vocal (AT2). Dans tous les cas, nous devons déterminer un alignement optimal et un score associé entre les représentations en API des requêtes et des documents. Nous avons pour cela adapté un algorithme de détection de mots dans un flot de parole continue (Gauvain *et al.*, 1982) lui-même basé sur un algorithme de programmation dynamique appliqué à la reconnaissance vocale (Sakoe *et al.*, 1978). La principale différence entre l'algorithme original de détection de mots de notre algorithme de détection de chaînes en API est de remplacer les vecteurs de caractéristiques audio (en général les « Mel Frequency Cepstral Coefficients » ou MFCCs) par des symboles de l'API.

3.1. *Minimisation de la distance d'édition*

À cause de fréquentes erreurs de transcription, soit dans les documents pour les deux tâches, soit dans les requêtes pour la tâche AT2, la recherche de la chaîne phonétique de la requête dans celle d'un document doit permettre une correspondance

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

inexacte. Que la correspondance soit exacte ou inexacte, il faut également lui attribuer un score afin de pouvoir classer en premier les documents pour lesquels la correspondance est la plus exacte. Afin de permettre les correspondances inexactes et d'attribuer un score à celles-ci, nous avons choisi de modifier la « distance » entre la chaîne de la requête et une sous-chaîne d'un document. Toutes les correspondances possibles entre la chaîne de la requête et l'ensemble des sous-chaînes d'un document sont prises en compte et, pour chacune de ces correspondances, une distance est calculée en comptant et en pénalisant l'ensemble des insertions, des suppressions et des substitutions entre la chaîne phonétique de la requête et la sous-chaîne du document. La figure 1 montre un exemple de calcul de distance.

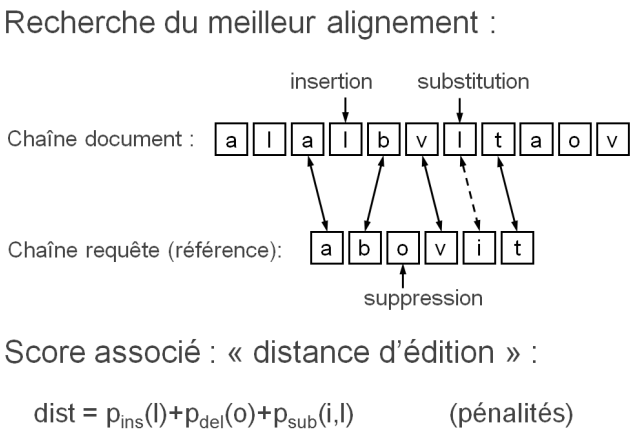

Figure 1. *Calcul du score d'une correspondance entre la chaîne de la requête et une sous-chaîne d'un document*

3.2. *Programmation dynamique*

La programmation dynamique est un moyen de résoudre le problème consistant à trouver le meilleur alignement et le score correspondant entre une chaîne requête et une chaîne document avec un temps de calcul linéaire avec la longueur de la chaîne requête et la longueur de la chaîne document.

Considérons la matrice produit de la chaîne de caractères représentant le document (horizontalement) et la chaîne représentant la requête (verticalement). Une correspondance (ou un alignement) valide entre la chaîne de la requête et une sous-chaîne du document est un chemin « continu et croissant » qui relie la rangée du bas de la matrice à la rangée du haut de la matrice (figure 2). Le meilleur alignement (ou chemin) est celui qui minimise la distance d'édition le long de lui-même. L'astuce de la programmation dynamique est de calculer le meilleur alignement par récurrence.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

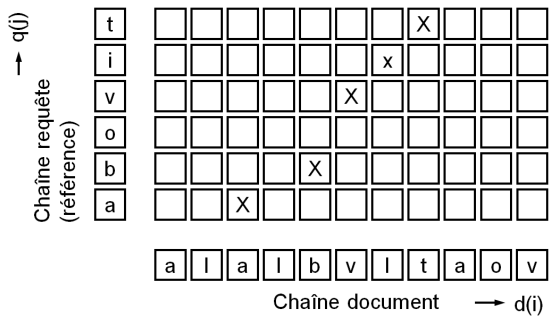

Figure 2. *Chemin d'alignement dans la matrice de programmation dynamique*

Si nous considérons la distance d'édition $e(i,j)$ selon le chemin optimal joignant la ligne du bas de la matrice au point (i,j) dans celle-ci, nous avons une équation de récurrence sur $e(i,j)$ car le chemin optimal arrivant en (i,j) doit :

– soit venir de $(i-2, j-1)$ avec une pénalité d'insertion,

– soit venir de $(i-1, j-2)$ avec une pénalité de suppression,

– soit venir de $(i-1, j-1)$ avec une pénalité de substitution éventuelle.

(à moins que l'un de ces points ne soit en dehors de la matrice).

$e(i,j)$ peut être calculé par récurrence dans la matrice complète en initialisant $e(i,j)$ à 0 sur la rangée du bas et à « infini » sur la colonne de gauche (à l'exception de la valeur du bas). L'équation de récurrence effectivement utilisée est donnée en équation 1. Les c_{xx} sont des constantes dont les valeurs sont : $c_{ii} = c_{dd} = 2.0$, $c_{sn} = c_{sd} = 1.0$, $c_{si} = 0.5$ (normalisation selon la requête de façon à ce que tous les alignements aient le même poids total et poids identiques pour les pénalités d'insertion, de suppression et de substitution).

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

Une fois terminé, le minimum de $e(i,j)$ sur la rangée du haut donne la meilleure distance d'édition qui est aussi le score du document pour la requête (le document avec le score le plus faible est le meilleur). Le retour en arrière à partir de la position du minimum donne l'alignement complet et donc la position dans le document de l'instance de la requête avec la meilleure correspondance.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

Le temps calcul de la distance d'édition par programmation dynamique entre une chaîne cible et une flux de symboles phonétiques, ainsi que la localisation de la meilleure occurence est linéaire en fonction de la longueur de la chaîne cible et de la longueur du flux dans lequel on la cherche. Comme on travaille ici avec des symboles qui sont des phonèmes dont le débit est très faible (une dizaine par seconde) et comme les calculs pour chaque point de programmation dynamique sont très simples, le temps de recherche d'une chaîne cible dans plusieurs dizaines d'heures de vidéo est de l'ordre de la seconde.

3.3. *Pénalités fixes et variables*

Les pénalités d'insertion, de suppression et de substitution peuvent soit être constantes soit dépendre des phonèmes effectivement insérés, supprimés ou substitués, certains phonèmes étant en effet plus susceptibles que d'autres d'être insérés, supprimés ou substitués. Pour les pénalités fixes, nous avons choisi :

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

et pour les pénalités variables, nous avons choisi :

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

Les probabilités ont été estimées par comparaison de transcriptions manuelles et automatiques.

4. Recherche basée sur le modèle vectoriel

Nous avons également fait des expériences avec le modèle vectoriel (MV), classiquement utilisé en recherche d'information textuelle. Nous avons utilisé des bigrammes de phonèmes comme unité d'indexation de base avec un système de normalisation à cosinus pivoté. Cette normalisation a été proposée par (Singhal *et al.*, 1996). Dans le modèle vectoriel de recherche d'information, on définit généralement dans le schéma de pondération, un facteur de normalisation visant à compenser le fait que de longs documents ont intrinsèquement plus de chances d'être pertinents pour une requête donnée. Une telle normalisation peut utiliser la norme du vecteur document ou la taille du document en termes de caractères, par exemple. (Singhal *et al.*, 1996) ont découvert que l'utilisation de la normalisation cosinus tend habituellement à trop stimuler les documents courts et à trop pénaliser les documents longs. C'est pourquoi ils ont proposé une normalisation pivotée pour compenser la normalisation habituelle. Dans notre cas, les documents varient beaucoup en longueur et une telle normalisa-

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

tion pivotée est néssaire. Par rapport à la pente de normalisation habituelle pour le texte (entre 0,2 et 0,3 selon (Singhal *et al.*, 1996), les pentes de la normalisation pour nos meilleurs résultats avec le modèle vectoriel sont de 0,4 pour AT1 et 0,54 pour AT2. Cette pente, qui correspond à une normalisation, est plutôt élevée ; ceci est probablement dû au fait que, sur cette collection, les documents longs on plus besoin d'être stimulés au cours de l'indexation pour être bien retrouvés.

5. Recherche par le contenu visuel

Nous utilisons la modalité visuelle pour la classification en concepts *via* un apprentissage supervisé exploitant les annotations fournies par les organisateurs du Star Challenge. Notre système de classification est générique dans la mesure où la même approche est développée pour détecter tous les concepts visés. L'approche met en œuvre des réseaux d'opérateurs qui incluent des extracteurs de descripteurs bas niveau, des détecteurs de concepts intermédiaires et des opérateurs de fusion (Ayache *et al.*, 2007) . Les sections suivantes décrivent ces étapes.

5.1. *Analyse visuelle*

Le flux visuel est analysé à plusieurs niveaux de granularité ; des descripteurs globaux représentent l'ensemble d'une image, tandis que des descripteurs locaux sont extraits dans des blocs entrelacés, selon une grille de N×M blocs. La participation au Star Challenge posant une contrainte de temps d'exécution, nous avons fixé la granularité par descripteurs empiriquement, de façon à obtenir des taux de classifications satisfaisant avec des temps de calculs réduits. Ces descripteurs sont utilisés pour l'apprentissage et la classification en concepts des séquences vidéo. L'analyse visuelle traite une à plusieurs images-clés par séquence vidéo, puis les combine selon les schémas de fusions classiques « précoce », « tardif », ou une combinaison des deux, puis après classification des images-clés, attribue un score par concept pour chaque séquence vidéo.

5.1.1. *Descripteurs bas niveau*

Nous considérons des descripteurs de couleur et texture globaux à l'image. La couleur est représentée par un histogramme 3D dans l'espace RGB, où l'espace de couleur est discrétisé de façon à obtenir un histogramme de 4×4×4 dimensions. La texture est extraite à l'aide de 40 filtres de Gabor sur 8 orientations et 5 échelles. Finalement, un descripteur visuel global est normalisé pour former un vecteur de 104 dimensions.

Pour compléter ces descriptions, nous extrayons d'autres descripteurs visuels dans chaque bloc d'image. Ces descripteurs sont alignés pour former une description visuelle riche pour chaque image clé :

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

Couleur (1) : décrit par un histogramme 3D de 3×3×3 dimensions, extraits dans une grille de 8×6 blocs. Ce descripteur forme un vecteur de 1 296 dimensions.

Couleur (2) : décrit par les deux premiers moments statistiques, extraits dans une grille de 8×6 blocs. Ce descripteur forme un vecteur de 432 dimensions.

Histogramme d'orientation : calculé dans une grille de 4×3 blocs. Chaque dimension correspond à la somme des magnitudes d'une orientation. Nous considérons 50 orientations. Le descripteur EDH forme un vecteur de 600 dimensions, il est connu pour être invariant en échelle et en translation.

Local Binary Pattern : calculé dans une grille de 2×2 blocs et constitue un vecteur de 1 024 dimensions. Le descripteur LBP modifie la valeur d'un pixel selon les valeurs des pixels voisins (3×3) pour capter des motifs de texture. LBP est invariant par une variation monotone de la valeur des pixels, ce qui est intéressant pour résister aux variations d'illumination (Mäenpää Topi, 2000).

5.1.2. *Descripteurs « sac-de-mots »*

La représentation d'images par sac-de-mots consiste à sélectionner un ensemble de régions dans une image (points d'intérêt) puis à décrire chacune d'elles à l'aide d'un descripteur visuel. Ces descripteurs sont alors quantifiés en affectant chaque descripteur à un élément d'un vocabulaire visuel pré-calculé. Cela permet d'obtenir un histogramme qui comptabilise les occurrences des mots visuels (éléments du vocabulaire visuel) dans une image. Combinée avec les descripteurs SIFT, invariants en échelle et en rotation, cette approche constitue l'une des approches les plus discriminantes pour la classification d'images (Lowe, 2004). Nous avons utilisé un dictionnaire de 1 000 mots visuels, fourni par le groupe INRIA-LEAR.

5.2. *Descripteur sémantique*

Ce descripteur vise à modéliser les relations sémantiques entre les concepts, par une approche sac-de-mots. Ce descripteur nécessite une phase d'apprentissage sur les blocs d'image. Pour cela, nous considérons chaque bloc d'image comme positifs relativement à un concept lorsque l'image est annotée positivement. Cette hypothèse est certes très forte mais peut être raisonnable pour certains concepts. Nous entraînons des modèles par concept au niveau bloc sur une partie de l'ensemble d'apprentissage puis classons l'ensemble des blocs, ce qui aboutit à $nb\_blocs \times nb\_concepts$ scores de classification par image. Le descripteur sémantique est représenté par un histogramme de $nb\_concepts$ dimensions où chaque dimension contient la somme des scores d'un concept sur tous les blocs.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

5.3. *Classification et fusion*

À partir des descripteurs visuels décrits précédemment, nous avons entraîné des classifieurs SVM à noyaux RBF pour la classification par concepts. Le choix des paramètres gamma et C par concept est fixé par validation croisée. Les différents descripteurs sont fusionnés par combinaison des schémas de fusion « précoce » et « tardif ». Une fusion précoce opère dans l'espace des descripteurs, tandis que la fusion tardive combine les scores de classifications obtenus par chaque classifieurs. Une combinaison de ces schémas de fusion est possible lorsque plus de deux descripteurs sont disponibles et apporte plus de flexibilité pour combiner les descripteurs. Par exemple, il est possible de fusionner séparément des descritpeurs couleur et texture de façon précoce, puis de fusionner les scores de classifications obtenus de chacun de façon tardive. De telles combinaisons améliorent significativement les performances de classification pour certains concepts.

Nous avons implémenté les opérateurs de fusions tels que l'opérateur de fusion précoce normalise les descripteurs et les aligne pour former un seul descripteur. L'opérateur de fusion tardive effectue une combinaison linéaire (moyenne) des scores de classification.

Pour notre participation au Star Challenge, nous avons mis en œuvre plusieurs réseaux d'opérateurs (*i.e.* plusieurs combinaisons de schémas de fusion, faisant intervenir différents descripteurs). Afin d'optimiser le choix du réseau optimal par concept visé, nous avons choisi, pour chaque concept, le réseau qui maximise la performance de classification sur un corpus de développement.

6. Recherche par le contenu multimodale

Les plans vidéo peuvent être évalués et triés en fonction de :

– la probabilité de présence d'un concept donné ;

– la similarité visuelle à une image ou à un plan vidéo donné ;

– la probabilité de présence d'une chaîne phonétique donnée.

Une requête mono ou multimodale peut être définie comme une combinaison de tels critères. Par exemple, dans les tâches vidéo 1 et 2 (VT1 and VT2) du Star Challenge, une requête est définie comme une combinaison d'un concept visuel requis et d'une similarité visuelle à une image (VT1) ou à un plan vidéo (VT2) donné. Dans les tâches de recherche multimodales 1 et 2 (AV1 and AV2), une requête est définie comme une combinaison d'une similarité visuelle à une image et de la présence d'une chaîne phonétique donnée textuelle (AV1) ou parlée (AV2).

Une approche similaire est utilisée dans tous les cas. Un score numérique est obtenu pour chaque critère en utilisant le sous-système approprié : recherche par chaîne phonétique, recherche par énoncé vocal, recherche par concepts pré-indexés ou re-

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

cherche par similarité visuelle à un exemple donné. Ces scores sont normalisés indépendamment pour chaque critère par une simple correction de moyenne et d'écart type. Une somme pondérée est ensuite calculée et celle-ci est utilisée pour trier les plans vidéo. Les poids optimaux sont choisis comme ceux qui maximisent la performance du système sur l'ensemble de développement.

La similarité visuelle est basée sur une distance Euclidienne sur les mêmes descripteurs de couleur, de texture et de mouvement que ceux qui sont utilisés pour la classification de concepts. La similarité visuelle est calculée séparément pour chaque caractéristique et les scores correspondants sont normalisés et combinés de la même façon que pour les composants multimodaux. Là encore, les poids optimaux sont choisis comme ceux qui maximisent la performance du système sur l'ensemble de développement.

7. Expérimentations

7.1. *Recherche phonétique monolingue, validation sur la collection de développement du Star Challenge*

L'objectif de la première série d'expériences était d'évaluer la performance relative des recherches basées sur une reconnaissance au niveau du mot et des recherches basées sur une reconnaissance au niveau du phonème ainsi que sur le bénéfice apporté par l'utilisation de pénalités variables dans le second cas. Trois méthodes basées sur la programmation dynamique (PD) sont comparées à quatre méthodes basiques de référence (*baselines*).

Ces expériences ont été menées sur la collection de développement audio du Star Challenge. Cette collection comprend environ deux heures (233 segments) de données audio monolingues (en anglais) et 39 requêtes résolues à la fois pour les tâches AT1 (requêtes par chaîne en API) et AT2 (requêtes parlées). Les systèmes doivent retourner une liste de 50 réponses et la mesure d'évaluation est définie comme le MAP du Star Challenge pour la recherche par l'audio (équation 2 ; ce MAP est différent de la norme TREC paramètres MAP) :

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

où L est le nombre total de requêtes, R_i est le nombre total de documents pertinents pour la $i$ème requête, et $\delta(i,j)$ est une fonction indicatrice qui vaut 1 pour les bonnes réponses (*i.e.* le $j$ème document pertinent est dans la liste de résultats pour la requête i) et 0 sinon.

Les documents (segments audio) et les requêtes AT2 ont été transcrits en API de deux façons. La première est une transcription au niveau mot (utilisant donc un

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

système de reconnaissance classique avec modèle de langage de mots) suivi par une conversion des mots en phonèmes. La seconde est une transcription directement au niveau phonétique (avec un système léger de décodage phonétique). Après cela et dans les deux cas, tous les documents et toutes les requêtes AT2 sont représentées par des chaînes en API.

Plusieurs méthodes basiques de référence ont été utilisées pour la comparaison. Une réponse aléatoire est un choix naturel et ceci constitue la « baseline 2 ». Une autre possibilité est de trier les segments en fonction de leur longueur, les segments les plus longs ayant le plus de chances *a priori* de contenir la requête. La « baseline 1 » correspond au choix des segments les plus courts (pire cas) et « baseline 3 » correspond au choix des segments les plus longs (meilleur cas). Ces trois références ignorent le contenu des documents (segments) comme celui des requêtes et les résultats sont les mêmes sur les tâches AT1 et AT2. La « baseline 4 » consiste en la recherche d'une présence exacte de la chaîne requête dans la chaîne document. Elle est équivalente à la commande Unix « grep » et aussi à une programmation dynamique avec des pénalités d'insertion, de suppression et de substitution infinies. Comme les correspondances exactes sont assez rares, les listes de résultats sont complétées par les segments restants les plus longs.

La programmation dynamique (PD) a été évaluée avec une reconnaissance au niveau du mot avec des pénalités fixes et variables et avec une reconnaissance au niveau phonétique avec des pénalités variables.

Finalement, le modèle vectoriel (MV) communément utilisé en recherche d'information a été essayé. Les termes d'indexation sont des bigrammes de phonèmes et le schéma de pondération avec une normalisation à cosinus pivoté a été utilisé.

Méthode	AT1	AT2
Baseline 1 : segments courts	0.024	0.024
Baseline 2 : hasard	0.242	0.242
Baseline 3 : segments longs	0.497	0.497
Baseline 4 : « grep » + segments longs	0.557	0.560
PD, rec. mot, pénalités fixes	0.776	0.632
PD, rec. mot, pénalités variables	0.843	0.636
PD, rec. phon., pénalités fixes	0.643	0.633
PD, rec. phon., pénalités variables	0.706	0.650
MV, rec. mot., bigrammes	0.797	0.660
MV, rec. phon., bigrammes	0.552	0.543

Tableau 2. *Validation sur la collection de développement du Star Challenge*

Le tableau 2 montre les résultats obtenus pour les méthodes testées et les méthodes de référence. Les observations suivantes peuvent être faites :

– les performances des baselines sont ordonnées comme prévu : courts $<$ hasard $<$ longs $<$ grep+longs ;

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

– les pénalités variables améliorent les performances de manière significative ;

– comme prévu également, la reconnaissance purement phonétique donne de moins bons résultats car elle ne bénéficie pas du modèle de langage au niveau mot ; elle est cependant la seule disponible pour les documents contenant des langues inconnues et c'est celle qui sera utilisée pour la recherche multilingue ; cette méthode est par ailleurs la plus performante sur la tâche AT2 (requêtes parlées) ;

– finalement, les systèmes basés sur le modèle vectoriel se comportent bien avec une reconnaissance au niveau du mot mais il fait légèrement moins bien que la baseline 4.

7.2. *Recherche phonétique monolingue, évaluation sur la collection « round 1 » du Star Challenge*

Le système ayant eu la meilleure performance sur la collection de développement a été utilisé pour la soumission officielle pour le « round 1 » du Star Challenge. Ce système utilise la programmation dynamique avec des pénalités variables. Une amélioration supplémentaire a été apportée ; elle consiste en l'utilisation de trois transcriptions différentes avec des poids différents affectés aux modèles de bigrammes de phonèmes, en faisant une moyenne des trois scores obtenus.

Collection	Pénalités	AT1	AT2	Moyenne
Dével.	fixes	0.760	0.679	0.719
Dével.	variables	0.858	0.728	0.793
Round 1	fixes	0.643	0.319	0.481
Round 1	variables	0.634	0.324	0.479

Tableau 3. *Influence de l'utilisation de pénalités variables*

Le tableau 3 montre les résultats obtenus par ce système sur la collection du « round 1 ». Cette collection est composée de 25 heures (4 300 segments) de documents audio monolingues (en anglais) et de 10 requêtes résolues pour les tâches AT1 et AT2. Les résultats correspondants sont aussi montrés pour le même système avec des pénalités fixes sur les données du round 1 et avec des pénalités fixes et variables sur les données de développement audio. Les observations suivantes peuvent être faites :

– les performances sur les données du round 1 sont très inférieures à celles obtenues sur les données de développement ;

– le gain de performance significatif obtenu par l'utilisation de pénalités variables sur les données de développement ne se retrouve pas sur les données du round 1 ;

– la chute de performance entre AT1 et AT2 est beaucoup plus importante sur les données du round 1.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

Tous ces effets sont probablement liés au fait que les données du round 1 contiennent beaucoup plus de segments dont une proportion beaucoup plus faible est pertinente, ce qui rend la tâche plus difficile. Les segments sont également plus courts. En utilisant cette approche, l'équipe LIG a terminé première sur AT1 et troisième sur AT2 parmi 35 équipes participantes.

7.3. *Recherche phonétique multilingue, validation sur la collection « round 1 » du Star Challenge*

Le but de cette série d'expériences est de valider la recherche multilingue en utilisant les données d'entraînement disponibles. Puisque les langues cibles ne sont pas connues, nous n'avons pu valider l'approche que sur les données monolingues (en anglais) disponibles. Nous avons toutefois construit des modèles en utilisant d'autres langues et nous les avons testés sur des données en anglais, en considérant que cela serait suffisamment représentatif. Nous avons d'abord essayé de construire des modèles à partir de langues uniques et de les utiliser pour l'indexation et la recherche : l'anglais (EN), le mandarin (CH) et le malais (MY). Nous avons aussi essayé un modèle acoustique multilingue qui est une combinaison des modèles de ces trois langues et du vietnamien (ML4). Les modèles de langage à base de unigrammes de phonèmes (1g) sont utilisés pour ces 4 premières expériences. Nous avons enfin essayé la même chose avec des modèles de langage à base de bigrammes phonétiques (2g) et une combinaison de trois transcriptions différentes avec des poids différents affectés aux modèles de bigrammes de phonèmes (combi3).

Modèles acoustiques	EN	CH	MY	ML4		
Modèles de langage	1g			1g	2g	2g (combi3)
AT1	0.668	0.476	0.428	0.603	0.612	0.650
AT2	0.585	0.578	0.577	0.568	0.579	0.638

Tableau 4. *Résultats obtenus avec différents modèles de langue*

Le tableau 4 montre les résultats obtenus avec ces différents modèles. Les observations suivantes peuvent être faites :

– nous avons utilisé un nouveau modèle de l'anglais qui s'est révélé être meilleur que celui que nous avions utilisé pour notre soumission officielle sur le round 1, en particulier sur la tâche AT2 ;

– les résultats obtenus avec les modèles de langage différents de celui de la langue cible (en mandarin ou en malais au lieu de l'anglais) sont très bons bien que les contenus phonétiques soient très différents ;

– les résultats sont meilleurs pour AT2 que pour AT1 dans ce cas, ce qui est probablement dû au fait que des confusions similaires sont faites au cours de la transcription des documents et des requêtes et qu'elles se compensent les unes les autres ;

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

– le modèle multilingue ML4 est presque aussi bon que le modèle purement anglais avec les modèles unigrammes de phonèmes et les modèles bigrammes font encore mieux. Ces modèles devraient par ailleurs, de par la façon dont ils sont construits, être aussi bon pour les langues asiatiques.

L'équipe LIG a utilisé le dernier système (ML4+2g avec une combinaison de 3 transcriptions) pour sa soumission officielle pour le round 3 et a obtenu avec celui-ci la cinquième place sur les tâches audio et la première place sur la tâche multimodale (recherche combinée audio et image de segments vidéo), se qualifiant ainsi pour la finale du Star Challenge à Singapour.

7.4. *Recherche visuelle, évaluation sur la collection « round 2 » du Star Challenge*

La tâche de recherche par le contenu visuel VT1 était de trouver des images (en pratique des images clés extraites de vidéos) qui contenaient un concept donné (parmi les 20 pour lesquels le système a été entraîné) et qui étaient visuellement semblables à une image donnée. Nous avons constaté que les meilleurs résultats ont été obtenus en pondérant dans un rapport 2 à 1 la similarité visuelle normalisée et la probabilité de présence du concept normalisée.

La tâche de recherche par le contenu visuel VT2 tâche était de trouver des plans vidéo qui contenaient un concept donné (parmi les 10 pour lesquels le système a été entraîné) et qui sont visuellement similaires à un plan vidéo donné. Nous avons encore constaté que les meilleurs résultats ont été obtenus en pondérant dans un rapport 2 à 1 la similarité visuelle normalisée et la probabilité de présence du concept normalisée. Cette approche nous a classés cinquièmes sur les tâches VT1 et VT2.

7.5. *Recherche multimodale, évaluation sur la collection « round 3 » du Star Challenge*

La tâche de recherche multimodale AV1 (resp. AV2) consistait à trouver des plans vidéo qui étaient visuellement similaires à une image donnée et qui contenaient une requête audio définie comme une chaîne en API (resp. comme un énoncé vocal). Nous avons constaté que les meilleurs résultats ont été obtenus en pondérant dans un rapport 3 à 7 la similarité visuelle normalisée et le score de détection de chaînes API normalisé. Cette approche nous a classés premiers sur les tâches multimodales AV1 et AV2.

Le tableau 5 montre les cinq premiers résultats trouvés pour la requête AV1 du « round 4 » fournie sous la forme de la chaîne phonétique « t e k n a : l i dZ i : » (probablement le mot *technology* en anglais imparfaitement transcrit). La chaîne trouvée apparaît dans une boîte avec un contexte d'une seconde avant et après. Parmi les cinq résultats, les quatre derniers sont bien des instances du mot *technology* et le premier

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

w Ou t w i n TH i : t i : m i k n a : l i dZ d f e : TH ^ f a j n @ n S ^ l
e n i : h a u t e k n a : l ^ dZ i : TH ^ t h @ z ^ p s e t
i N h e v ^ l i : i n TH i : n j u : t e k n a : l ^ dZ i : z h w ^ n g i d i g z @ m
@ n d S e r w i TH TH e m a : n t e k n a : l ^ dZ i : z
n h i : S u d g i t TH ^ l ei t ^ s t t e k n a : l ^ dZ i : h w i tS i

Tableau 5. *Les cinq premiers résultats trouvés pour la requête AV1 fournie sous la forme de la chaîne phonétique « t e k n a : l i dZ i : »*

est un faux positif (qui a peut-être été sélectionné pour l'aspect visuel du segment, les requêtes AV étant en fait multimodales).

8. Conclusion

Nous avons présenté une approche fondée sur l'utilisation de l'alphabet phonétique international (API) pour la recherche selon le contenu de vidéos multilingues. Une telle approche peut fonctionner même si les langues parlées dans les documents sont inconnues. Notre technique a été validée dans le contexte du « Star Challenge », une compétition de recherche d'information organisée par l'agence singapourienne A-STAR. L'approche présentée inclut la construction d'un modèle acoustique multilingue à large couverture, contenant des unités API, et la proposition d'une méthode de recherche fondée sur la programmation dynamique. La programmation dynamique permet de repérer la chaîne de la requête dans la chaîne du document, même avec un taux d'erreur de transcription au niveau phonétique significatif. Les méthodes que nous avons développées nous ont classés premiers et troisièmes sur les tâches de recherche monolingues (anglais), cinquièmes sur la tâche de recherche multilingue et premiers sur la tâche de recherche multimodale (audio et image).

Les résultats obtenus montrent le potentiel d'une telle approche fondée sur l'API pour indexer et retrouver des documents audiovisuels dans une langue inconnue. Des expériences complémentaires seraient nécessaires sur de plus grands corpus pour confirmer cette tendance. Des améliorations seraient par ailleurs possibles au niveau de la qualité des modèles multilingues et de la recherche fondée sur l'alignement dynamique qui pourrait être améliorée en exploitant des graphes d'hypothèses (treillis) en sortie du système de décodage phonétique.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com

Enfin, la combinaison de l'audio (API), de l'indexation par concepts et de la similarité visuelle, s'est avérée efficace pour la tâche de recherche d'information selon le contenu de vidéos multimodales.

Remerciements

Ce travail a été en partie soutenu par le programme Quaero.

9. Bibliographie

Ayache S., Quénot G., « Image and video indexing using networks of operators », *J. Image Video Process.*, vol. 2007, n° 4, p. 1-13, 2007.

CCC, « http ://www.dear.com/CCC/resources.htm », 2005.

Clarkson P., Rosenfeld R., « Statistical Language Modeling using the CMU-Cambridge Toolkit », *Eurospeech'07*, p. 2707-2710, 1997.

Gauvain J.-L., Lee C., « Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains », *IEEE Trans. on Speech and Audio Processing*, vol. 2, n° 2, p. 291-298, 1994.

Gauvain J.-L., Mariani J.-J., « A method for connected word recognition and word spotting on a microprocessor », *Proc. IEEE ICASSP 82*, vol. 2, p. 891-894, 3-5 May, 1982.

LDC, « http ://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp ?catalogId=LDC93S6B », 1993.

LDC, « http ://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp ?catalogId=LDC98S71 », 1997.

Le V.-B., Do-Dat T., Castelli E., Besacier L., Serignat J.-F., « Spoken and written language resources for Vietnamese », *LREC'04*, p. 599-602, 2004.

Lowe D., « Distinctive image features from scale-invariant keypoints », *International Journal of Computer Vision*, vol. 60, p. 91-110, 2004.

Mäenpää Topi Pietikäinen Matti O. T., « Texture classification by multi-predicate local binary pattern operators », *15th International Conference on Pattern Recognition*, vol. 3, p. 951-95, 2000.

Moraru D., Besacier L., Meignier S., Fredouille C., Bonastre J.-F., « Speaker Diarization in the ELISA Consortium over the last 4 years », *RT2004 Fall Workshop*, 13-14 Nov., 2004.

Placeway P., Chen S., Eskenazi M., Jain U., Parikh V., Raj B., Ravishankar M., Rosenfeld R., Seymore K., Siegler M., Stern R., Thayer, « The 1996 Hub-4 Sphinx-3 System », *In DARPA Speech Recognition Workshop*, Chantilly, VA, February, 1997.

Sakoe H., Chiba S., « Dynamic programming algorithm optimization for spoken word recognition », *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, n° 1, p. 43-49, 1978.

Singhal A., Buckley C., Mitra A., « Pivoted document length normalization », *ACM SIGIR conference*, ACM Press, p. 21-29, 1996.

Stolcke A., « SRILM – an extensible language modeling toolkit », *Intl. Conf. on Spoken Language Processing*, 2002.

Cet article des Editions Lavoisier est disponible en acces libre et gratuit sur dn.revuesonline.com