Structuration automatique de flux télévisés

Gaël MANSON et Sid-Ahmed BERRANI

Orange Labs - France Telecom R&D
4, rue du Clos Courtel. BP 91226
35510 Cesson-Sévigné. France.
{gael.manson, sidahmed.berrani}@orange-ftgroup.com

Résumé – Les flux télévisés sont des documents audiovisuels bien structurés : ils sont en effet composés de programmes successifs (films, séries, documentaires, jeux, journaux, ...). Dès que les flux sont diffusés sur les ondes, ils perdent malheureusement toute information de structuration. La problématique est de retrouver automatiquement cette structuration, c'est-à-dire le début et la fin de chaque programme, à partir des signaux audiovisuels reçus. L'originalité de la solution est de proposer un système complètement automatique basé uniquement sur l'analyse du flux vidéo. Pour cela, le système utilise les propriétés relationnelles et contextuelles des répétitions des inter-programmes (bandes annonces, publicités, sponsors). Le flux est d'abord segmenté à partir d'une détection des séquences visuelles répétées puis les segments sont classés en programmes ou inter-programmes (IP) grâce à la programmation logique inductive. Finalement, les segments de programmes forment la structuration du flux. Les résultats présentés montrent une précision temporelle bien meilleure que celle indiquée dans les guides de programmes TV fournis par les chaînes de télévision.

Abstract – TV streams are well structured audiovisual documents : they consist of consecutive television programs (movies, serials, documentary films, games, news, ...). As soon as they are broadcasted on the air, the TV streams lose unfortunately information on their structure. The problem consists in automatically recovering, from the received linear audio/visual signal, the original structure of the TV stream, i.e. the start and the end of each broadcasted program. We propose a novel solution that is only based on the video stream analysis. Our technique makes use of the relational and contextual information of the repeated inter-programs (commercials, trailers, sponsors) in the stream. A first segmentation step based on a near-identical repeated video sequence detection is performed. Resulting segments (the occurrences of repeated sequences and the rest of the stream) are then classified in program or inter-program (IP). The proposed solution for that is based on Inductive Logic Programming. Finally, program segments constitute the TV stream structuration. Our solution achieves a temporal imprecision that outperforms the electronic program guides provided by the TV channels.

1 Introduction

Les chaînes de télévision fournissent aujourd'hui du contenu en continu et leur nombre ne cesse de croître. En France, le fond de l'Institut National de l'Audiovisuel (INA) chargé d'archiver les diffusions françaises augmente de 540 000 heures par année et au final plus de 4 000 000 heures de programmes y sont consultables [1]. Face à ce volume de données audiovisuelles, de nouveaux besoins et services sont apparus :

- l'archivage de ces données ;
- le contrôle des diffusions ;
- la pige de publicités ;
- l'accès de façon non linéaire au contenu souhaité, c'est-à-dire sans contrainte de l'heure de diffusion.

L'ensemble de ces services repose sur une structuration des flux audio-visuels, c'est-à-dire une segmentation pour en extraire les programmes (films, séries, documentaires, jeux, journaux, ...) et les inter-programmes (bandes annonces, sponsors et séquences de publicité en particulier) diffusés en continu puis une annotation ou classification de ceux-ci. Ces traitements étant extrêmement couteux lorsqu'ils sont réalisés manuellement, des techniques automatiques sont nécessaires afin d'exploiter le grand volume de flux TV disponible.

Les chaines de télévision fournissent généralement des métadonnées décrivant leur flux télévisuel. Ces métadonnées peuvent être diffusées avec le flux (c'est le cas pour l'*Event Information Table*, EIT) ou bien disponibles sur internet (comme pour l'*Electronic Program Guide*, EPG). Elles contiennent un guide des programmes diffusés avec leurs horaires et leurs descriptions. Bien que les chaînes de télévision connaissent les informations exactes de structuration lors de la production, les métadonnées associées aux flux télévisés ne contiennent que des horaires approximatifs. L'étude menée dans [1] montre l'imprécision et l'incomplétude de ces métadonnées.

Dans ce papier, nous présentons un système automatique pour la structuration des flux audiovisuels en segments de programmes et en segments d'inter-programmes sans utiliser de métadonnées. Nous exposons d'abord l'art antérieur puis nous détaillons notre système basé sur la programmation logique inductive. Enfin, nous présentons nos résultats.

1. http://www.ina.fr/entreprise/activites/depot-legal-radio-tele

2 État de l'art

Il s'agit d'un problème récent qui a déjà fait l'objet de quelques études. De manière générale, si l'on ne dispose pas d'exemples de structuration sur plusieurs années [2], l'approche de structuration généralement utilisée consiste à détecter les inter-programmes (IPs). Les IPs sont les séquences audiovisuelles courtes (bandes annonces, publicités, jingles) qui séparent deux parties d'un même programme ou deux programmes consécutifs. Ces IPs partagent de nombreuses propriétés communes. En particulier, les IPs sont diffusés plusieurs fois dans le flux. Ces propriétés font que les IPs sont beaucoup plus faciles à détecter que les programmes longs. Ceux-ci sont quant à eux hétérogènes (séries, documentaire, film, émissions) et ne partagent en général pas de propriétés communes.

Les méthodes de détection d'IP peuvent se diviser en trois catégories :

1. méthodes basées sur des caractéristiques particulières. Ces méthodes utilisent des caractéristiques propres aux IPs comme par exemple : les silences, les images monochromes, ou les absences de logos [5]. Cependant, elles sont généralement spécifiques à un seul type d'IPs : les publicités ;
2. méthodes basées sur des bases de références. Ces méthodes stockent dans une base de références les IPs étiquetés et classés. Les IPs sont ensuite reconnus dans le flux grâce à cette base de références [7]. Des techniques basées sur des signatures vidéo ou audio peuvent être utilisées pour cette reconnaissance [8]. L'inconvénient principal de ces approches est que la base de références doit être créée et mise à jour régulièrement à la main ;
3. méthodes basées sur les répétitions. Ces méthodes utilisent le fait que les IPs sont diffusés régulièrement presque à l'identique dans le flux. Les séquences audiovisuelles répétées peuvent être détectées en utilisant [3] une table de hachage et le contenu vidéo, en étudiant [4] la corrélation du contenu audio d'un morceau du flux sur une portion plus longue de flux, ou en effectuant [2] un *clustering* des images clés similaires du flux.

Les méthodes basées sur les répétitions sont les plus prometteuses car elles n'utilisent que la propriété de répétition des IPs sans base de références et ne sont pas spécifiques à une chaîne TV particulière. Par exemple, sur la chaîne France 2, l'absence de publicités sur la chaîne depuis janvier 2008 rend impossible l'utilisation des images monochromes. Par contre, les programmes restent séparés par des bandes annonces qui se répètent régulièrement.

Les approches basées répétitions présentées sont les plus adéquates pour la structuration automatique. Néanmoins, elles ne sont capables que de détecter des répétitions dans le flux. Certaines de ces répétitions sont effectivement des IPs mais d'autres sont des portions de programmes. Par exemple, un reportage sur une actualité importante sera répété entre les différents journaux de la journée. Les génériques des émissions seront aussi répétés. C'est pourquoi il est nécessaire de classer les segments résultants de la segmentation basée répétitions. Cette classification est l'objet de ce travail. Elle est présentée dans la suite.

2. J.-P. Poli [9] utilise la relative stabilité de la programmation des chaînes de TV et apprend leur structuration à partir des bases de données des dernières années de structuration. Cette approche nécessite une très grande base de données audiovisuelles qui est très coûteuse à établir.

3 Structuration automatique de flux télévisuels

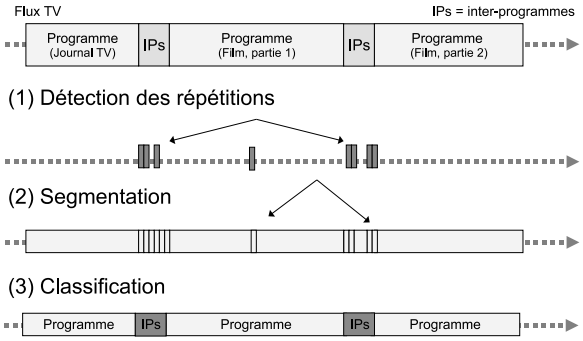

FIGURE 1 – Schéma général de notre solution de structuration.

L'objectif de notre solution est de segmenter le flux en segments de programme et en segments d'IPs sans utiliser de bases de références ou des métadonnées. Le schéma général de notre solution de structuration est présenté dans la figure 1 :

1. les répétitions sont détectées selon la méthode de [2] ;
2. le flux est segmenté à partir des répétitions, chaque occurrence des répétitions forme un segment et chaque écart entre les segments forme un autre segment ;
3. les segments identifiés sont classés en programmes ou IPs.

Dans notre solution, le flux télévisé est enregistré et accumulé en continu. Puis régulièrement les répétitions sont calculées. Ensuite, le flux est segmenté et les segments sont classés grâce aux répétitions. Le système fournit donc une structuration du flux télévisuel de manière périodique.

Pour le problème de classification des segments, le rôle des relations de voisinage est important. Contrairement aux techniques d'apprentissage qui essayent de trouver des caractéristiques audiovisuelles de segments pour définir une publicité ou une bande annonce, nous considérons que la place d'un segment et les caractéristiques des segments voisins importent beaucoup dans la détermination d'un segment.

L'originalité de notre solution est d'utiliser les informations contextuelles et relationnelles des répétitions pour la classification. Une description détaillée de cette approche est présentée dans [6]. Chaque segment est décrit en utilisant trois groupes de caractéristiques :

1. les caractéristiques locales. Elles sont propres à un seul

segment. Nous utilisons la durée et le nombre de répétitions du segment ;

2. les caractéristiques contextuelles. Par exemple, la moyenne du nombre de répétitions du segment suivant chaque occurrence de la répétition du segment. Nous utilisons aussi les caractéristiques locales et contextuelles des segments voisins ;

3. les caractéristiques relationnelles. Celles-ci sont les classes attribuées aux segments voisins.

Afin d'utiliser au maximum les informations relationnelles et contextuelles, nous avons utilisé la programmation logique inductive (PLI) pour modéliser les caractéristiques locales, contextuelles et relationnelles. L'idée principale est d'apprendre sur un ensemble d'apprentissage des règles de la forme suivante :

« *le segment A est un IP si A dure entre 25 et 35 secondes et si A se trouve entre deux IPs* ».

L'ensemble de ces caractéristiques est donc décrit comme des faits encodés en logique du premier ordre. La PLI induit et généralise ensuite au maximum des règles logiques reliant tous ces faits logiques à partir d'exemples positifs et d'exemples négatifs. Une fois les règles apprises, nous les appliquons suivant un protocole précis pour classer automatiquement nos segments. Ce protocole applique en premier les règles utilisant les caractéristiques locales, puis celles employant les caractéristiques contextuelles, puis enfin, celles employant les caractéristiques relationnelles et ce de manière récursive. Nous avons utilisé *Aleph* [10] comme système de PLI. *Aleph* (ex *P-Progol*) permet une PLI descendante en partant de règles logiques très générales qu'il spécifie.

Parmi les règles logiques apprises, certaines sont génériques et ainsi très pertinentes. Cependant, d'autres règles restent spécifiques à l'ensemble d'apprentissage. Afin d'évaluer le niveau de pertinence d'une règle, un ensemble de validation est utilisé. Cet ensemble de validation contient de nouveaux exemples positifs et négatifs à partir desquels le système teste la pertinence des règles apprises. Le protocole de classification est alors modifié afin d'appliquer en premier les règles les plus pertinentes puis les autres règles triées par niveau de pertinence.

Lors de la phase d'apprentissage et de validation, la PLI utilise des exemples positifs et négatifs. Ces exemples sont issus d'une segmentation et d'un étiquetage manuels du flux et les métadonnées ne sont pas utilisées. Les ensembles d'apprentissage et de validation sont spécifiques à chaque chaîne et un nouvel apprentissage est requis pour chaque chaîne à traiter.

En plus de bien modéliser les informations relationnelles et contextuelles, la PLI permet de manipuler des règles explicites pour la classification. Ces règles peuvent être facilement complétées par des règles *a priori* données par un utilisateur expert.

4 Évaluation expérimentale

Pour évaluer notre structuration automatique en programmes et IPs, nous avons utilisé un flux TV de 6 jours issu d'une chaine publique française. Nous avons créé une vérité terrain (VT) sur cette semaine en déterminant manuellement le début et la fin de chaque programme et de chaque IP. Sur cette période de 6 jours nous nous sommes intéressés plus particulièrement à la période de 18h à minuit. Cet ensemble a été découpé en 2 jours d'apprentissage, 1 jour de validation et 3 jours de test.

Nous avons effectué dans un premier temps une segmentation automatique basée répétitions. Puis nous avons classé les segments issus des répétitions suivant notre solution. Nous avons détecté 798 segments pour les ensembles d'apprentissage et de validation et 805 segments pour l'ensemble de test. La PLI a permis d'apprendre un ensemble de 35 règles logiques. Des exemples des règles apprises les plus pertinentes sont donnés dans La figure 2. Nous présentons les résultats de la classification dans la colonne *PLI* du tableau 1.

	PLI	$\leq$ 3 min	VT
Précision (%)	93.51	70.01	96.43
Rappel (%)	97.12	98.66	98.63

TABLE 1 – Évaluation de la classification en IP basée PLI.

Afin de mesurer les performances de notre classification, nous avons calculé la précision et le rappel au niveau plan. La précision est dans ce contexte le nombre de plans correctement classés comme IP sur le nombre total de plans classés comme IP par le système. Le rappel est ici le nombre de plans correctement classés comme IP sur le nombre total de plans appartenant à des IPs d'après la VT.

Les résultats présentés montrent une bonne classification des segments en IP [3]. À titre de comparaison, le tableau montre les résultats en ne considérant comme IPs que les segments de durée inférieure à 3 minutes (colonne $\leq$ *3 min*). Ce seuil de 3 minutes a été choisi empiriquement. Il permet de rapatrier tous les IPs (le rappel est proche de 100 %). Le tableau présente aussi les résultats (colonne *VT*) obtenus en utilisant directement la classification manuelle de la VT. Ces résultats basés VT servent à montrer les limitations de notre approche. Nos résultats sont proches de ces limites.

Comme expliqué dans l'introduction, la structuration automatique n'est qu'une étape vers l'élaboration de nouveaux services et applications. Une des applications principales est la délinéarisation automatique des programmes du flux TV qui consiste à délimiter précisément le début et la fin de chaque programme diffusé. Cette application ouvre la voie aux services de télévision à la demande et offre ainsi la possibilité aux téléspectateurs de regarder leurs programmes favoris sans contraintes de temps. Notre système de structuration automatique permet de séparer les IPs du reste des programmes. Appliqué sur une chaine publique (dans laquelle les programmes ne sont pas coupés par des publicités), notre système permet

3. Le reste des segments est alors classé en programme.

$$IP(A) \leftarrow non(NbRepetitions(A, 0)), NbPrecedentesRepetitions(A, 10\_20).$$
$$P(A) \leftarrow NbRepetitions(A, 2), NbPrecedentesRepetitions(A, 0), NbSuivantesRepetitions(A, 0).$$
$$IP(A) \leftarrow Dist(A, B, 1), IP(B), NbRepetitions(A, 10\_20).$$
$$P(A) \leftarrow Dist(A, B, -1), P(B), NbRepetitions(A, 0).$$

Avec :
– $IP(A)$: *Le segment A représente un inter-programme*
– $P(A)$: *Le segment A représente un programme*
– $Dist(A, B, N)$: *Le segment B est à une distance de N segments du segment A*
– $NbRepetitions(A, c/a\_b)$: *Le segment A se répète c fois/entre a et b fois*
– $NbPrecedentesRepetitions/NbSuivantesRepetitions(A, c/a\_b)$: *La moyenne du nombre de répétitions du précédent/suivant segment de chaque occurrence de la répétition du segment A est c fois/entre a et b fois*

FIGURE 2 – Exemples de règles logiques apprises par la PLI possédant un haut niveau de pertinence.

d'extraire les programmes diffusés. Nous montrons dans le tableau 2, l'application de la classification pour la délinéarisation sur l'ensemble de test précédemment utilisé. L'objectif est ici de retrouver à la seconde près les segments de programmes. Pour cela, les segments consécutifs de classe programme sont fusionnés et nous évaluons la précision temporelle (moyenne μ et écart type σ) au début et à la fin par rapport à la vérité terrain. Afin d'évaluer nos résultats, nous présentons à titre de comparaison les résultats obtenus en utilisant les métadonnées de l'EPG. Notre solution automatique est de manière significative bien plus précise que l'EPG.

	Début		Fin	
	μ	σ	μ	σ
Not. Sol.	5.6 s	12.8 s	11.7 s	17.7 s
EPG	2 m 14.0 s	1 m 4.2 s	4 m 6.5 s	3 m 48.2 s

TABLE 2 – Précision temporelle de l'extraction de programmes sur une chaîne publique.

5 Conclusion

Dans cet article, nous avons présenté notre approche pour la structuration automatique de flux télévisuels en programmes et inter-programmes. La contribution concerne plus particulièrement la classification des segments en segments d'inter-programmes en utilisant des caractéristiques contextuelles et relationnelles à travers la programmation logique inductive. A partir de cette structuration, nous avons aussi montré que l'on pouvait extraire les programmes TV de manière très précise.

Une des perspectives intéressantes de ces travaux consiste à classer les inter-programmes en publicités, bande-annonces, sponsors ou jingles suivant la même approche basée sur la PLI. L'objectif serait alors d'apprendre des règles relationnelles et contextuelles qui différencient une bande-annonce d'une publicité ou d'un sponsor. Nous envisageons aussi d'étudier une manière automatique de fusionner les différentes parties d'un même programme séparées par des coupures publicitaires. Cette réunification de parties de programmes permettrait une délinéarisation automatique sur les chaînes privées.

Références

[1] S.-A. Berrani, P. Lechat, and G. Manson. TV broadcast macro-segmentation : Metadata-based vs. content-based approaches. *Proc. of the ACM Int. Conf. on Image and Video Retrieval, Amsterdam, Pays-Bas*, juillet 2007.

[2] S.-A. Berrani, G. Manson, and P. Lechat. A non-supervised approach for repeated sequence detection in TV broadcast streams. *Signal Processing : Image Communication, spec. iss. on "Semantic Analysis for Interactive Multimedia Services"*, pages 23(7)525–537, 2008.

[3] J. M. Gauch and A. Shivadas. Finding and identifying unknown commercials using repeated video sequence detection. *Jour. of Computer vision and image understanding*, 103(1) :80–88, 2006.

[4] C. Herley. Argos : automatically extracting repeating objects from multimedia streams. *IEEE Trans. on Multimedia*, pages 8(1) :115–129, 2006.

[5] R. Lienhart, C. Kuhmunch, and W. Effelsberg. On the detection and recognition of television commercials. *Proc. of the IEEE Int. Conf. on Multimedia Computing and Systems, Ottawa, Ontario, Canada*, juin 1997.

[6] G. Manson and S.-A Berrani. An inductive Logic Programming-Based Approach for TV Stream Segment Classification. *Proc. of the IEEE Int. Symp. on Multimedia, Berkeley, Californie, USA*, décembre 2008.

[7] X. Naturel, G. Gravier, and P. Gros. Fast structuring of large television streams using program guides. *Proc. of the Int. Work. on Adaptive Multimedia Retrieval, Genève, Suisse*, juillet 2006.

[8] J. Oostveen, T. Kalker, and J. Haitsma. Feature extraction and a database strategy for video fingerprinting. *Proc. of the Int. Conf. on Recent Advances in Visual Information Systems, Hsin Chu, Taiwan*, mars 2002.

[9] J.-P. Poli and J. Carrive. Modeling television schedules for television stream structuring. *Proc. of ACM Int. MultiMedia Modeling Conf., Singapore*, janvier 2007.

[10] A. Srinivasan. Aleph : A learning engine for proposing hypotheses. `http://web2.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.pl`, 2007