Indéxation et recherche par le contenu de documents vidéos

Adrien Brilhault
Magistère 2 Informatique
UFR IMA
60, rue de la Chimie
38041 GRENOBLE

Catherine Garbay
(directrice de stage)
Equipe MAGMA
LIG, 385 avenue de la Bibliothèque
38400 Saint Martin d'Hères

Georges Quénot
(directeur de stage)
Equipe MRIM
LIG, 385 avenue de la Bibliothèque
38400 Saint Martin d'Hères

***Résumé*—Ce rapport traite de l'indexation sémantique de documents vidéos, ou en d'autres termes de la reconaissance de concepts de haut-niveau par des méthodes de classification basées sur un apprentissage supervisé. Nous avons examiné différentes pistes de recherche en vue de l'amélioration des modèles et systèmes existants, pour finalement nous focaliser sur la création de méthodes automatiques de sélection et de fusion des différentes sources d'information dans la chaine de traitement de l'image.**

***Mots-cléfs* —Selection automatique de descripteurs, Recherche d'images par le contenu, Multimédia, Descripteurs, Classification, Concepts, Recherche d'information, Apprentissage supervisé, Fusion.**

I. Introduction

La production de contenus multimédia a été multipliée dans des proportions considérables ces dernières années, notamment en ce qui concerne les images et les vidéos (plusieurs milliards de photos indexées par Google ou Flickr, et environ 65 millions de vidéos disponibles sur YouTube). La recherche d'images, et plus généralement de contenus multimédias, est donc un besoin qui va en se généralisant avec la production croissante de documents. Cette tâche s'inscrit d'ailleurs dans un grand axe de recherche européen, le programme Quaero, fédérant des projets sur l'analyse de données et de contenus multimédia.

La disponibilité de grandes sources d'images annotées permet maintenant d'envisager plusieurs directions de recherche combinant la sémantique des images et leur aspect visuel pour l'identification plus fine des éléments présents dans les images. Nous observons ainsi une convergence des recherches textuelles et basées sur le contenu. Des services de recherche d'images sont maintenant proposés par la plupart des grands moteurs de recherche. Si historiquement ces systèmes n'exploitaient que le texte environnant l'image dans la page web, ils incorporent maintenant quelques caractéristiques propres à l'image, essentiellement sur son format (résolution de l'image, standard vs. panoramique, couleur vs. noir & blanc, photo vs. graphique,...) mais également sur leur contenu (comme la détection de visages récemment intégrée à Exalead et Google).

Les résultats issus de ces travaux sur la recherche d'images par le contenu ont donné lieu à une prolifération de systèmes qui commencent à être mis en oeuvre sur des grandes collections telles que Flickr, un comparatif détaillé d'une quarantaine de moteurs de recherche d'images a été publié en 2002 dans [1], depuis cette étude on peut également citer Idée, Behold ou enfin PicItUp, proposant tous une recherche d'image sémantique (par concepts).

A. Contexte

Dans le but de parcourir, rechercher et manipuler des documents vidéos, des index décrivant leur contenu sont nécessaires. Afin d'élargir le champ d'utilisation possible de ces vidéos par d'autres logiciels, nous souhaiterions disposer d'une information la plus riche et complète possible. Classification et indexation sont donc deux faces d'une même préoccupation : rapprocher "physiquement" les éléments semblables en vue d'une visualisation ou d'une recherche efficace.

Cependant, ce travail d'indexation a jusqu'à maintenant été majoritairement réalisé par des documentalistes, annotant manuellement le contenu des extraits vidéos à partir d'un certain nombre de mots-clés.

Cette tâche s'avère très coûteuse et face à l'explosion de la quantité de contenus disponibles on conçoit aisément qu'elle ne peut être généralisée à grande échelle, une classification automatique s'avère donc nécessaire. Cette méthode communément appelée indexation de vidéos[1] consiste à assigner automatiquement des étiquettes relatives au contenu sémantique des documents vidéos.

B. Systèmes de recherche d'information

Les premiers systèmes d'indexation et de recherche d'images par le contenu étaient uniquement basés sur une description des documents en terme de caractéristiques bas-niveau (histogrammes de couleur, texture, formes, dimensions, etc). Ces systèmes permettaient par exemple de rechercher dans une base des images similaires à une image exemple fournie par l'utilisateur, et retournaient donc en sortie non pas une classe d'appartenance, mais un certain nombre d'images jugées pertinentes et similaires à l'image requête proposée. Une autre méthode de recherche dans ces systèmes dits de première génération consistait pour l'utilisateur à formuler sa requête directement à partir de ces caractéristiques bas niveau que nous avons cité (couleur, texture,...)

Le but premier d'un tel système étant de fournir aux utilisateurs des outils efficaces de recherche et de navigation, il est donc nécessaire de prendre en compte les besoins et le comportement d'un utilisateur humain. Or il est difficile pour un individu lambda de formuler une requête en termes de descripteurs bas-niveau. On conviendra qu'il est plus intuitif d'exprimer une attente lors de la recherche d'un document multimédia par un ensemble de mots-clés qu'en terme de d'histogrammes de couleur ou de magnitude de gradients ! De ces besoins ont émergé les systèmes de deuxième génération.

Cette nouvelle vague de systèmes visent à l'indexation sémantique des images et des vidéos, afin d'offrir à l'utilisateur la possibilité de rechercher des documents à l'aide de concepts ou de mot-clés tel que cela est proposé depuis de nombreuses année en recherche d'information textuelle. Ces concepts sémantiques permettant de décrire le contenu de l'image ou de la vidéo peuvent être de natures

[1] Notion introduite par Hampapur *et al.*, sous le nom "Feature based digital video indexing" [2].

diverses, et donc de niveaux d'abstraction différents. Ils peuvent aussi bien représenter un objet que des lieux, des actions, des sujets thématiques, ou bien encore des personnes.

La grande difficulté dans le cadre de la recherche d'images est l'extraction de ces informations "sémantiques". Dans le domaine textuel il est possible de rechercher directement un concept au sein d'un document, et ainsi de pouvoir renvoyer à l'utilisateur formulant par exemple la requête "voiture" les documents contenant ce mot. En revanche comment déterminer si une image contient bien une voiture si l'on ne dispose pas de méta-données sur celle-ci (annotations ou texte entourant l'image) ? C'est la tout l'enjeu de le recherche par le contenu dans des bases d'images ou de vidéos.

Cette distance entre d'un part le signal brut de l'image ou de la vidéo et de l'autre les concepts sémantiques qui la définissent est appelée le *fossé sémantique*. Ce terme introduit par Eakins *et al.* est défini dans [3] : le franchissement du fossé sémantique consiste à inférer des caractéristiques haut-niveau, nécessitant un certain degré de raisonnement logique, à partir des informations primitives qu'une machine est capable d'extraire d'une image, telle que ses couleurs ou sa texture.

C. Problématique

Avec les avancées faites ces dernières années en vision par ordinateur, l'analyse d'images et la reconnaissance d'objets (ou de concepts) commence à être considéré comme un domaine mature, où de moins en moins de systèmes sont construit "from scratch". Comme le souligne B. Draper dans *"Adaptative Object Recognition"* [4], la plupart des plateformes développées de nos jours sont au contraire mises en place en chainant différents modules standards de vision tels que le lissage d'image, des techniques d'*enhancement* , d'extraction d'arêtes, de segmentation de régions, d'association de formes, de structures symboliques, de calcul de flots optiques,... Ceux-ci sont couplés à d'autres composants non spécifiques au traitement d'image, tels que des classifieurs, des opérateurs de fusion, des techniques d'optimisation de paramètres, etc.

La sélection, l'enchainement et le paramétrage de ces différents modules sont en général basés sur l'intuition des développeurs et leur connaissance du contexte d'application, puis raffinés par essais/erreurs. Or rien ne permet de conclure sur la pertinence de ces heuristiques et d'estimer leur optimalité.

Nous proposons donc dans ce rapport différentes méthodes en vue d'une combinaison optimale et automatisée des sources d'information dans un processus de classification d'images, et ceci de manière adaptée à chaque concept, à partir des données d'apprentissage disponibles. Les conclusions de Koen Van de Sande [5] dans son évaluation des descripteurs pour la reconnaissance d'objets vont également dans ce sens, suggérant que, si les performances de descripteurs pris individuellement ont souvent été étudiées, l'analyse des stratégies de combinaison automatique reste une voie assez peu explorée qui mériterait l'attention.

Les résultats prometteurs des récent travaux de de Marsalek *et al.*, utilisant des algorithmes génétiques pour la pondération des différentes sources [6] laissent aussi penser que la mise en place de nouveaux schémas de fusion utilisant des stratégies de sélection automatique pourraient accroître les performances des système de recherche d'images par le contenu.

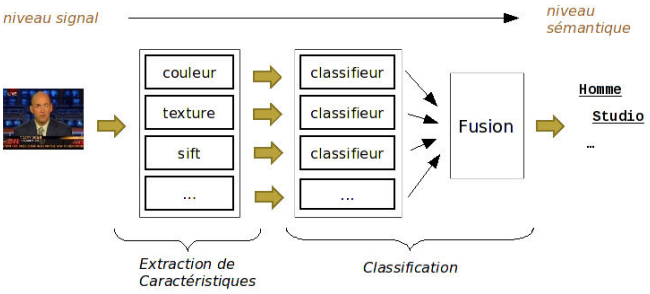


FIG. 1. Architecture standard des systèmes d'indexation d'images par le contenu

II. ÉTAT DE L'ART

La classification d'images est une tâche permettant de déterminer par exemple la présence d'objets (avions, chaises, voitures, etc), d'évènements (manifestations, tremblements de terre), ou encore de scènes (environnement urbain, studio, plage,...). La grande majorité des systèmes de recherche d'images ou de vidéos utilisent pour cela une méthode d'apprentissage supervisé[1] consistant à entraîner des classifieurs à partir de données extraites des documents multimédia par des descripteurs.

Lorsque différentes sources d'informations sont utilisées, dans le cas par exemple où plusieurs descripteurs sont appliqués aux images ou vidéos, une dernière étape consistant à combiner ces différentes informations par des opérateurs de fusion est alors nécessaire. Le pipeline communément utilisé est donc résumé dans la figure 1.

A. Métriques d'évaluation

Commençons tout d'abord par introduire les différentes métriques d'évaluation que nous utiliserons tout au long de ce mémoire. La recherche d'images par le contenu[2] est un sous-domaine de la recherche d'information. Les mesures d'évaluation adoptées ont donc logiquement été reprises de la RI classique. Les deux informations les plus utilisées sont la précision et le rappel.

- La précision correspond au pourcentage de documents pertinents retournés par rapport au nombre total de documents renvoyés :

  $$Précision = \frac{|Pertinents \cap Retournés|}{|Retournés|}$$

- Et le rappel est défini par le nombre de documents pertinents retournés par rapport à tous des documents pertinents de l'ensemble d'évaluation :

  $$Rappel = \frac{|Pertinents \cap Retournés|}{|Pertinents|}$$

Il a été montré que ces deux mesures sont inversement corrélés. En effet le rappel augmente généralement avec le nombre de documents retournés, alors que dans le même temps la précision a tendance à chuter. Il est donc rare d'obtenir a la fois un bon score de rappel et une haute précision. Pour combiner ces deux valeurs on utilise généralement la précision moyenne (ou average precision[3]), qui correspond à l'aire en dessous de la courbe de rappel-précision.

Soit $l^k = \{d_1, d_2, ..., d_k\}$ une liste ordonnées de k documents d'un ensemble de test T. Si R est l'ensemble des documents pertinents de

[1] Qui sera défini plus précisément dans la section II-C.
[2] Ou *Content Base Image Retriaval*, que l'on notera par la suite *CBIR*.
[3] Généralement abrégé par *AP*.

T, nous pouvons calculer le nombre de documents pertinents parmi les k premiers éléments de l par $|R \cap l^k|$ et définissons la fonction ψ telle que $\psi(d_k) = 1$ si $d_k \in R$ et 0 sinon. La précision moyenne est alors donnée par :

$$AP(l) = \frac{1}{|R|} \sum_{k=1}^{|T|} \frac{|R \cap l^k|}{k} \psi(d_k)$$

Dans des campagnes comme TRECVID, où l'évaluation des résultats des participants est faite sur un corpus de très grande taille (plus de 35000 documents multimédias), la grande difficulté dans le calcul des performances des différents systèmes est de disposer de jugements de pertinence pour chacun des documents. Cette annotation manuelle a un coût trop important pour pouvoir être mis en oeuvre lors de l'évaluation. Un *"pooling"* est donc réalisé, et seule une partie des documents est finalement jugée par les annotateurs. Il s'agit alors d'estimer les performances à partir de jugements incomplets. Différentes méthodes ont été proposées, celle utilisée par TRECVID, l'*Inferred Average Precision* (ou *infAP*), a montré de très bons résultats. Les détails relatifs au calcul de l'infAP sont donnés dans [7].

Lorsque ces mesures (AP standard ou infered AP) sont calculées pour plusieurs classes (dans notre cas différents concepts), on défini alors la mesure MAP (Mean Average Precision) comme la moyenne arithmétique des AP calculées individuellement pour chacune des classes.

B. Descripteurs

L'extraction de caractéristiques constitue le premier pas de toutes les procédures d'analyse d'images qui visent à un traitement symbolique de leur contenu. Les descripteurs bas-niveau fournissent en effet une première représentation symbolique à partir du signal brut de l'image (les pixels), et constituent donc une étape majeure dans le franchissement du fossé sémantique.

Les éléments de base de la plupart des descriptions symboliques d'images sont les points, les arêtes et les régions [8]. De très nombreuses méthodes existent pour décrire l'image en terme de caractéristiques bas niveau, constituant autant d'angles d'interprétations possibles du contenu de l'image. Parmi les différents systèmes de reconnaissance de concept proposés lors de la dernière campagne TRECVID le nombre de descripteurs utilisés pouvait aller de un jusqu'à plus d'un millier ! Les principales approches sont résumées dans cette section.

1) Descripteurs globaux: Une description statistique globale des caractéristiques d'une image est une technique très utilisée dans l'analyse d'image pour l'indexation et la recherche de documents. Ces attributs globaux sont facilement calculables et réussissent souvent à capturer une information pertinente sur le contenu de l'image. Bien qu'ils ne permettent pas d'évaluer la distribution spatiale des caractéristiques de l'image et donc sa structure interne, leur importance ne doit pas être sous-estimée. Selon le contexte leur aide peut être précieuse, rechercher les images contenant un grand nombre de lignes droites est par exemple un bon critère dans la détection des constructions humaines. De manière générale les descripteurs globaux peuvent apporter d'importants indices sur l'apparence visuelle générale d'une image, son type, ou certaines autres propriétés. Nous avons regroupé les principaux descripteurs globaux en deux grandes classes détaillées ci dessous, ceux permettant de représenter les informations de couleur d'une part, et ceux décrivant la texture de l'image.

▷ *Couleur*

Il existe de nombreuses façons de modéliser l'information donnée par les couleurs mais la plus répandue est l'utilisation d'histogrammes de couleurs [9], qui fournissent la distribution globale des couleurs dans l'image. Plusieurs variantes existent, dépendant notamment du choix de l'espace de couleurs utilisé (RGB, HSV, YCrCb, Color-Oppoent, RG, ou HMMD), se référer aux articles [5] et [10] pour un descriptif plus complet de ces méthodes et l'étude de leurs propriétés d'invariance aux changements d'intensité et de couleur de la lumière.

Les histogrammes de couleurs ne modélisant pas la distribution spatiale des couleurs, les moments de couleurs [11] sont une alternative permettant d'incorporer à différents degrés des informations sur la répartition spatiale des couleurs. L'idée derrière cette approche est que toute distribution de couleur peut être caractérisée par ses moments. De plus, comme les informations les plus importantes sont concentrées dans les premiers moments, le descripteur peut se contenter d'extraire les moments de premier ordre (moyenne), deuxième et troisième ordre (variance et asymétrie). Les moments de couleur généralisés M_{pq}^{abc} d'ordre $p+q$ et de degré $a+b+c$ ont été défini par Mindru *et al.* [12] par la formule suivante, où I est la fonction associant à chaque point de l'image de coordonnées (x, y) la valeur du pixel pour chacune des composantes de couleur ($I : (x, y) \mapsto (R(x, y), G(x, y), B(x, y))$) :

$$M_{pq}^{abc} = \iint x^p y^q [I_R(x, y)]^a [I_G(x, y)]^b [I_B(x, y)]^c \, dx \, dy$$

En plus des histogrammes de couleurs et des moments de couleurs on peut également mentionner quelques autres descripteurs souvent abordés dans la littérature : les *Color Sets* [13], les *Color Coherence Vector* [14], les histogrammes de corrélation de couleur[1] [15] ou encore les descripteurs SCD (Scalable Color Descriptor) [10], Dominent Color [16] et CSD (Color Structure Descriptor) [10].

▷ *Texture*

Dans les années 70 Hararalick *et al.* ont proposé une des premières méthodes de caractérisation des textures baptisée *Co-Occurrence Matrix of Texture Feature* [17]. Cette approche explorait les dépendances spatiales des textures en construisant d'abord une matrice de co-occurrence basée sur l'orientation et la distance entre les pixels de l'image puis en représentant la texture par l'extraction de statistiques sur cette matrice, comme le contraste, l'entropie ou la différence inverse des moments.

De nos jours les filtres de Gabor [18] sont souvent reconnus comme les descripteurs les plus efficaces pour représenter textures et surfaces. Ils permettent la détection de contours et motifs selon différentes orientations et échelles. L'image est découpée en blocs réguliers, pour lesquels sont calculés les moyennes et déviations standard de l'énergie des pixels.

Parmi les autres descripteurs de texture mentionnons les *Local Binary Patterns* [19], basés sur une analyse multi-résolution des niveaux de gris, ou encore les modèles markoviens (*Markov Random Field Representation* [20]), le filtrage multi-source, et les représentations basées sur les fractales (*Fractal-Based Descriptors* [21]).

[1] Color Correlograms.

2) Descripteurs de régions: On parle souvent de descripteurs de régions lorsque l'on applique des descripteurs "globaux" sur un sous-échantillonnage de l'image. Ils traitent donc au final l'image originale de façon "locale", mais sont à différencier des descripteurs locaux qui font généralement référence aux approches basées sur la détection de points d'intérêt dans l'image. Pour les descripteurs de régions[1] l'analyse "locale" est le résultat d'une division de l'image en blocs réguliers, généralement caractérisée par 2 paramètres : le nombre de patchs en X et en Y, qui peuvent aussi dans certains cas être complétés par la taille des patchs.

Les descripteurs globaux sont alors appliqués sur chacune des régions de manière indépendante (pour le descripteur, chaque bloc est traité comme une image différente), puis les vecteurs de caractéristiques extraits de chaque patch sont ensuite concaténés en un vecteur final. A titre d'exemple si on prend le cas d'un histogramme de couleur RGB classique de dimension 3, après avoir divisé l'image en 8x6 régions, et calculé l'histogramme de chacun des 48 blocs on obtiendrait au final un vecteur de caractéristiques à 144 dimensions.

3) Descripteurs locaux: Comme nous l'avons expliqué les descripteurs d'images globaux, bien que performants pour une grande quantité de requêtes, ont néanmoins certaines limitations. Ils sont beaucoup moins efficaces lorsque, par exemple, on recherche des images d'un concept donné prises dans des conditions de vue très différentes. La distribution globale des couleurs et textures change alors radicalement d'une image à une autre, et une description plus locale est alors requise, pour capturer la structure interne de l'image, d'une manière robuste aux changements de point de vue ou d'illumination.

Les descripteurs locaux apportent cette robustesse aux occlusions et aux changement de conditions de vue (lumière, bruit, point de vue), ainsi que l'invariance aux rotations ou aux changements d'échelle. Ils sont communément considérés comme les méthodes offrant les meilleures performances et de plus ne nécessitent pas de segmentation de l'image. Ces descripteurs locaux sont généralement calculés en deux étapes. La première consiste à identifier des points d'intérêt dans l'image, puis les valeurs du descripteur sont ensuite produites en utilisant les caractéristiques de l'image autour de chacun de ces points d'intérêt précédemment trouvés. La détection de ces points est généralement effectuée à différentes échelles, obtenues en sous-échantillonant l'image initiale.

Il existe une grande variété de descripteurs locaux (voir [22]), qui diffèrent par exemple dans la façon de calculer les points d'intérêt, tâche des *détecteurs*, tels que les détecteurs de Laplace, Hariss, Susan, Laplacian of Gaussian, Forstner,.... En plus des différences dans le choix du détecteur les descripteurs locaux se distinguent également par le nombre de points retenus et la façon de traiter l'image autour de ceux-ci. Le rôle du descripteur est de caractériser l'apparence locale de l'image autour des points identifiés. De ces différents paramètres dépendent les propriétés d'invariance mentionnés précédemment tels que l'invariance au changement d'échelle, à la rotation, à la translation ou au changement de point de vue. Une étude des principaux descripteurs locaux est proposée dans [23] et [24], détaillant notamment les SIFT [25], PCA-SIFT [26], GLOH [24] , Steerable Filters [27], Differential Invariants [28], ou encore les Shape Context Descriptor [29].

[1] On parle également de descripteurs par "patchs".

C. Classifieurs

Afin de déterminer les caractéristiques visuelles des classes d'objets que nous cherchons à détecter il est nécessaire d'avoir un ensemble d'images annotées, c'est à dire une base de référence pour laquelle nous est donnée la présence ou l'absence des concepts dans chacun des documents, qui nous permettra d'apprendre ces classes à partir de leurs exemples. Ceci nous positionne donc dans une démarche d'apprentissage supervisé, par opposition à l'apprentissage non-supervisé qui consiste à regrouper un ensemble de données non annotées en groupes homogènes, ceux-ci ne donnant par ailleurs aucune information sur leur interprétation "sémantique".

En apprentissage supervisé, la classification consiste à estimer une fonction $y = f(X)$ à partir d'un ensemble d'exemples de la forme $\{(X_1, y_1), \ldots, (X_n, y_n)\}$, les valeurs de y appartenant à ensemble fini de classes $\{1, ..., K\}$, représentant dans notre cas les concepts à détecter. La fonction apprise est appelée un classifieur. L'apprentissage est donc réalisé sur un ensemble de couples (X_i, y_i) exprimant la présence du concept $y_i \in \{1, \ldots, K\}$ dans le i^{eme} document de la base d'apprentissage. Les variables X_i sont typiquement des vecteurs de la forme $\langle x_{i,1}, x_{i,2}, \ldots, x_{i,dim}\rangle$ dont les composantes sont des valeurs réelles caractérisant l'échantillon i (appelées *features* ou *caractéristiques* de X_i). Ces vecteurs sont généralement extraits par les descripteurs bas-niveau, que nous avons introduits dans la section précédente.

1) Support Vector Machines: Les SVM[2] sont des méthodes de classification binaire par apprentissage supervisé qui furent introduite par Vapnik en 1995 [30]. Cette méthode conçue pour une séparation de deux ensembles de données repose sur l'existence d'un classificateur linéaire dans un espace approprié.

Pour des données linéairement séparables il existe une infinité d'hyperplans séparant sans erreurs ces ensembles. L'hyperplan optimal, appelé *hyperplan à marge maximale* est celui situé à la distance maximale des vecteurs les plus proches parmi la base d'exemples (voir figure 2). Le but des SVMs est de trouver cet hyperplan maximisant la marge de séparation entre deux classes. Nous ne détaillerons pas ici les algorithmes d'optimisation de la recherche des hyperplans, qui sont fournis dans le livre de Vapnik.

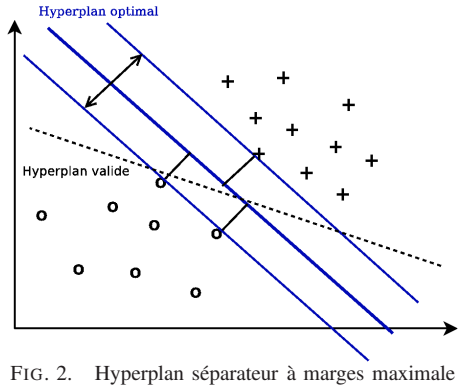


FIG. 2. Hyperplan séparateur à marges maximale

Dans la plupart des problèmes rencontrés en pratique, les données ne sont cependant pas séparables par un hyperplan. Il convient alors de modifier l'approche précédente afin de prendre en compte la possibilité d'observations mal classées. Pour permettre aux SVM de répondre à ces problèmes complexes de classification, les algorithmes

[2] Support Vector Machines, en français Machines à Vecteurs de Support, ou Séparateurs à Vastes Marges.

initiaux ont été transformés pour élaborer des structures de détection non-linéaires. L'extension au cas non-linéaire peut être effectuée en transformant les observations à l'aide d'une fonction ϕ puis en appliquant un détecteur linéaire. La fonction ϕ est implicitement définie par le choix d'un noyau K tel que $K(x_i, x_i') = \langle\ \phi(x_i); \phi(x_i')\ \rangle$ où x_i est une observation tirée de la base d'apprentissage χ. La fonction K est donc définie ainsi :

$$\begin{aligned} K : \quad \chi \times \chi &\longrightarrow \mathbb{R} \\ (x_i, x_j) &\longmapsto K(x_i, x_j) \end{aligned}$$

Les noyaux permettent donc d'étendre aisément au cas non-linéaire des techniques d'apprentissage initialement développées pour le cas linéaire. Le choix du noyau et de ses paramètres se fait généralement d'une manière heuristique lors de tentatives du type essai-erreur. La recherche de méthodes reposant sur des arguments théoriques solides n'en est pas moins un des défis à relever dans l'avenir des méthodes à noyaux. Une liste exhaustive de noyaux reproduisants et des développements supplémentaires sur ces noyaux, comme par exemple la combinaison de noyaux, peut être consultée dans [30].

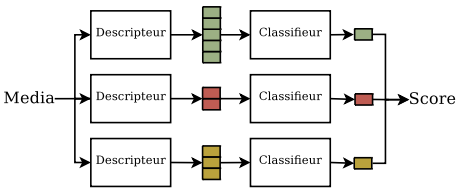
2) Autres classifieurs: Dans le domaine de la recherche d'images et de vidéos par le contenu la plupart des systèmes se tournent ces dernières années vers des solutions basées sur les classifieurs SVM. D'autres approches de classification restent néanmoins utilisées par de nombreuses équipes, comme les k plus proches voisins, généralement noté KNN, les modèles de Markov, les réseaux de neurones ou encore les réseaux bayésiens.

D. Fusion d'informations

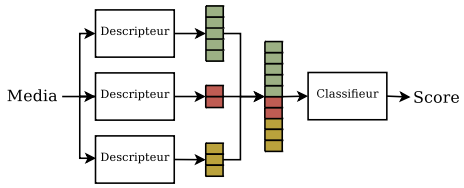
La fusion d'informations vise à combiner différentes sources de données, que constituent les diverses méthodes d'extraction de caractéristiques des images (les différents descripteurs).

Dans les approches probabilistes, une autre manière de voir la fusion consiste à considérer que chaque source donne une probabilité d'appartenance à une classe, et que la fusion consiste à combiner ces probabilités pour trouver la probabilité globale d'appartenance à la classe. Cette vision revient à considérer la fusion comme un problème d'estimation et permet d'utiliser des opérateurs de combinaison différents du produit. En particulier les méthodes de moyenne ou moyenne pondérée, de médiane ou de consensus sont souvent employées.

1) Fusion tardive: Les schémas classiques de fusion tardive consistent à entraîner des classifieurs pour chacun des descripteurs individuellement puis à combiner les scores issus de chacun afin d'obtenir un unique score final de prédiction. Différentes fonctions sont applicables pour la fusion des scores unimodaux, comme le produit, la somme, le Maximum,... Ces méthodes sont détaillées plus amplement dans [31], proposant aussi d'autres schémas de fusion tardives tels que les systèmes de vote.



2) Fusion précoce: La fusion précoce consiste à regrouper les caractéristiques issues des différents descripteurs avant le processus d'apprentissage [32]. Comme la dimension des vecteurs de caractéristiques varie d'un descripteur à l'autre une simple concaténation des différents vecteur aura pour effet de favoriser les descripteurs de grande dimension. Pour remédier à ce biais une normalisation des valeurs de chaque vecteur peut être effectuée avant leur concaténation.



3) Fusion de rangs: La fusion de rangs est un cas particulier de fusion tardive. Au lieu de traiter individuellement chaque document en fusionnant les scores de prédiction des différents classifieurs, on considère plutôt son rang parmi l'ensemble des autres documents du corpus de test. Ainsi chaque image ou vidéo reçoit la moyenne des rangs prédits par les différents classifieurs, puis ces scores globaux sont finalement utilisés pour reclasser l'ensemble des documents. On peut alors utiliser pour cette fusion de rangs une moyenne arithmétique, une moyenne géométrique, ou enfin une moyenne harmonique[1]. Cette dernière variante est généralement considérée comme le meilleur choix pour une fusion de rangs[2].

E. Réseaux d'opérateurs

La plupart des systèmes de détection de concepts dans des bases d'images ou de vidéos sont des combinaisons des descripteurs et classifieurs des sections précédentes, organisés selon une architecture en pipeline descripteur-classifieur-fusion. Ils consistent à extraire des descriptions de bas niveau de chaque modalité, puis à entraîner le système pour reconnaître un concept donné. Un algorithme d'apprentissage supervisé modélise alors un concept visé en terme des régularités identifiées dans les descripteurs de bas niveau.

Le fossé sémantique est donc dans un premier temps pris en charge par des algorithmes d'analyses du signal, puis, une grande part est laissée à l'algorithme d'apprentissage. Ce type d'architecture pose de nombreuses contraintes et laisse peu d'espace de liberté pour améliorer les performances d'un système.

Pour s'affranchir de ces contraintes, Stéphane Ayache propose une représentation du fossé sémantique comme un espace continu [34]. Il pose ainsi l'hypothèse d'une continuité entre les données numériques et conceptuelles, en passant par divers niveaux d'abstraction, et qu'il est donc possible de franchir le fossé sémantique par parties.

L'architecture qu'il propose se base sur un réseau d'opérateurs organisés en flots de données, chacun des opérateurs permettant de combler une partie du fossé sémantique, à l'opposée des approches classiques où un unique opérateur est généralement en charge du passage du signal numérique aux descriptions sémantiques. Les flots de données circulant entre ces opérateurs sont baptisés *numcepts* et visent à établir une continuité entre les données numériques et conceptuelles. Ils permettent ainsi une homogénéisation des données manipulées pendant le processus d'indexation et donc de s'abstraire

[1] La moyenne harmonique correspond à l'inverse de la moyenne arithmétique de l'inverse des termes.

[2] Les équipes MRIM et LIRIS à TrecVID 2008 ont cependant constaté de meilleures performances en utilisant une fusion arithmétique [33].

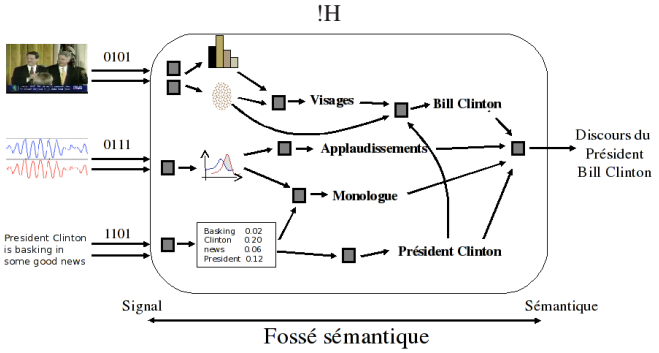


FIG. 3. Le fossé sémantique et le continuum de numcepts [35]

des modalités. Un numcept peut contenir différents types d'informations, qu'il s'agisse de données au niveau signal, à des niveaux intermédiaires, ou au niveau sémantique. Les numcepts sont alors mis en relation les uns avec les autres dans un réseau afin d'exploiter différentes formes de contexte et de permettre l'inférence ou la dérivation de nouveau concepts. Les opérateurs peuvent donc être définis comme des modules du réseau prenant en entrée des numcepts, et produisant en sortie d'autres numcepts [35].

La figure 3 illustre cette organisation des données et le continuum entre l'espace signal et l'espace sémantique. Les boites carrées symbolisent les opérateurs et les flèches représentent les flux de données, c'est à dire les numcepts, qui peuvent être de simples pixels, des chaines de caractères, des contours de région, des formes, des concepts, des sujets, etc. Une telle architecture a l'avantage d'être souple : il est possible d'envisager de nombreuses variantes de réseaux, par l'agencement des opérateurs, et par la nature de ces opérateurs. Par ailleurs, la problématique de fusion des modalités peut être envisagée à différents niveaux d'abstraction : il s'agit de combiner des numcepts.

III. CONSTRUCTION AUTOMATIQUE DES RÉSEAUX D'OPÉRATEURS

Notre étude se positionne dans le prolongement des travaux de Stéphane Ayache que nous venons de présenter. Il avait développé une approche à base d'opérateurs organisés en flots de données comme solution flexible au franchissement du fossé sémantique dans des systèmes d'indexation [35]. Les données manipulées aux différents niveaux du processus de détection des concepts "sémantiques" sont modélisées par des *numcepts*, produits par différents opérateurs, et le système de détection de concepts est alors construit en combinant ses différents opérateurs en réseaux.

Cependant ces réseaux étaient jusqu'à'a maintenant créés manuellement, résultant d'intuitions sur les combinaisons susceptibles de montrer de bon résultats, et de phases d'expérimentations permettant de valider ou non ces hypothèses. De plus ils étaient génériques, le même réseau d'opérateurs étant appliqués a chacun des concept avec les mêmes paramètres.

Or nous pensons que des concepts différents requièrent des réseaux spécifiques, et notamment l'utilisation de descripteurs différents, afin d'illustrer cette idée prenons par exemple le concept "Ciel". Un descripteur comme les histogrammes de couleurs sera évidement plus adapté à la détection de ce concept qu'une méthode reposant sur les points d'intérêt telle que les SIFT. En revanche pour un concept comme "Voiture" la situation sera inverse, les couleurs globales de l'image n'apporteront que peu d'indices sur la présence ou non d'une voiture, alors que les SIFT permettrons de reconnaitre ses motifs caractéristiques.

De plus, comme l'ont montré Van De Sande *et al.*, la pertinence d'un descripteur pour un concept donné dépend non seulement de sa nature, c'est à dire du type d'information qu'il fourni sur l'image (sa couleur, sa texture, ses formes, etc), mais également de ses propriétés d'invariance [5]. Ainsi des descripteur robustes aux changement d'échelle ou autres transformations affines (rotation, translation,...) seront particulièrement efficaces pour des concepts de type "objet" (personne, vélo, table), et moins pour des "scènes" (mer, montagne), où l'on privilégiera des descripteurs invariants aux changements d'intensité et de couleur de la lumière.

Ces différents constats montrent la nécessité de spécialiser les réseaux selon les concepts et les contextes. Leur construction suppose donc l'acquisition de connaissances concernant le choix des descripteurs, leur importance dans le réseau, et ces réseaux ne peuvent donc pas à priori être construit manuellement. Une construction automatique semble donc nécessaire en vue de combinaisons optimales menant à une meilleure gestion de la richesse des informations disponibles et donc à de meilleures performances finales dans la détection des concepts.

En raison de la grande difficulté de poser à priori une méthodologie d'extraction de connaissances pour un domaine aussi vaste, nous avons opté pour une démarche exploratoire et expérimentale qui permet d'appréhender le problème dans ses différentes dimensions.

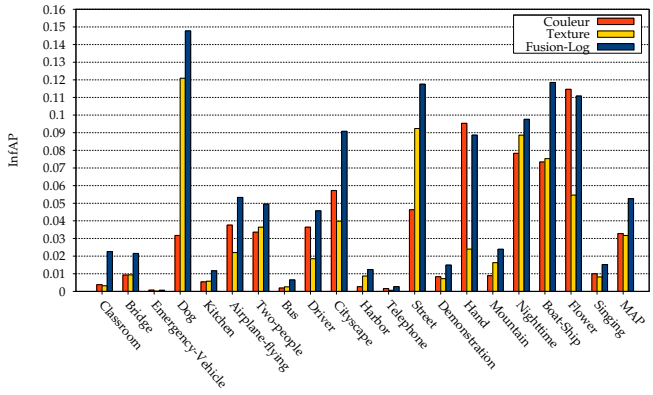
Nous avons donc d'abord cherché à déterminer si pour une paire de descripteurs il existait des heuristiques de combinaison permettant de prendre en compte l'importance relative de chaque descripteur en vue d'une fusion optimale, ceci fera l'objet de la partie III-B. Puis dans la section III-C nous avons mis en place une méthode de construction de réseaux inspirée des algorithmes de sélection séquentielle de caractéristiques. Nous proposerons enfin deux autres façons de combiner un ensemble de descripteurs, la première basée sur les performances individuelles de chaque opérateur d'extraction, et la deuxième sur une optimisation globale de leur poids.

A. Analyse préliminaire

Lors des premières expérimentations que nous avions conduites au début de la mise en place de la plateforme expérimentale nous avons obtenu les résultats reportés dans la figure 4. Ces premiers runs consistaient en un réseau basique, constitués de deux descripteurs bas niveau (*glcolor* et *imvec*, fournissant respectivement des informations globales sur la couleur et la texture de l'image). Deux classifieurs SVM ont donc été entraînés pour chacun des concepts à partir de ces caractéristiques extraites des données annotées. Leurs scores sur les plans du corpus d'évaluation ont ensuite été combinés par une fusion tardive linéaire.

On peut tirer plusieurs constats de ces premiers résultats. Tout d'abord on remarque que les performances individuelles et relatives des deux descripteurs varient fortement d'un concept à un autre. Ainsi on dénote d'une part une différence dans la difficultés des concepts (certains obtiennent de "bon" scores de classification, d'autres sont très faibles), et d'autre part que selon le concept un descripteur sera plus pertinent que l'autre pour identifier la classe. Enfin, nous observons des résultats plutôt surprenant de la fusion des deux descripteurs. Si dans certains cas la fusion des deux sources d'informations mène à un gain important (voir supérieur à la somme des performance de chacune, par exemple pour le concept "Classroom"), dans d'autre

FIG. 4. Evaluation des performances de deux descripteurs et de leur fusion



cas elle conduit à des performances inférieures au meilleur des deux descripteurs (pour les concepts "Hand" et "Flower").

Ceci nous permet donc de valider notre hypothèse initiale, à savoir la nécessité de spécialiser les réseaux par concept afin de sélectionner des descripteurs appropriés pour chacun, mais également d'adapter automatiquement les méthodes de fusion, de sorte qu'il n'en résulte pas de baisse des performances comme c'est le cas ici. Il nous faut donc rechercher des heuristiques en vue d'une pondération optimale des descripteurs au niveau de la fusion.

B. Prédiction d'une fusion optimale pour une paire de descripteur

Lors d'une fusion de descripteurs, il s'agit de combiner les informations fournies par chacun (typiquement les scores de prédiction), en vue d'une estimation générale de la confiance en une prédiction. Différentes méthodes de fusion peuvent être mise en oeuvre, telles que celles présentées dans la section II-D, cependant en plus du choix du type de fusion à appliquer, la pondération des différentes sources constitue un paramètre important à prendre en compte, car comme nous l'avons vu précédemment une pondération mal choisie peut mener à une dégradation des performances. Or dans les travaux précédents réalisés au sein de l'équipe cette pondération n'était utilisée, tous les descripteurs se voyant affecter un même poids.

Cette fusion sans pondération montre des résultats très inégaux selon les concepts et les descripteurs, tel que nous l'avons observé dans l'analyse préliminaire, et dans l'optique de mettre en place une construction automatique de réseaux par *sélection séquentielle avant* nous nous somme d'abord intéressé à la fusion de deux descripteurs.

Nous cherchions à déterminer si des heuristiques pouvaient être appliquées au choix des poids de deux descripteurs lors de leur fusion, c'est à dire estimer une fonction permettant de prédire un poids pour un couple de descripteurs, basée sur leur ratio de performances. Les informations obtenues sur les performances des descripteurs ne pouvant être calculées sur le corpus d'évaluation, car nous ne disposons pas d'information sur la présence des concepts dans ces plans, nous avons donc divisé le corpus d'apprentissage, une partie étant utilisée pour l'entraînement des classifieurs, et l'autre pour évaluer leurs prédictions et donc obtenir un estimation de leurs performances pour chacun des concepts. Nous avons ensuite estimé cette fonction de prédiction de poids par une régression linéaire. Pour cela nous avons mesurer les résultats de la fusion (la précision moyenne) sur un grand nombre d'échantillons (toutes les paires de descripteurs possibles, sur chacun des 20 concepts, soit 900 échantillons, en faisant varier les poids de -1 à 1 par incréments de 0.1).

Nous avons effectué cette régression pour différents sous-ensembles du corpus d'apprentissage et également sur le corpus de test à titre de comparaison, et l'on observe peu de variation dans les coefficients directeurs calculés, ce qui nous permet de conclure sur la stabilité de cette heuristique face au changement de corpus.

Cette fonction issue de la régression linéaire sur le corpus d'apprentissage nous permet donc maintenant d'estimer directement un poids supposé optimal pour une paire de descripteurs, évitant ainsi d'avoir, à chaque fusion, à évaluer les performances de nombreuses combinaisons de poids pour déterminer celles conduisant à de bons résultats.

Afin d'évaluer l'apport de cette méthode nous avons comparé sur le corpus de test, pour chaque échantillon, les résultats d'une fusion sans pondération avec ceux utilisant la prédiction des poids issue de la régression effectuée sur le corpus d'apprentissage. A titre d'indication du gain maximum que l'on peut espérer nous avons également comparé ces performances à celles de la fusion optimale de chacune des paires de descripteurs, que nous avons mesurées sur le corpus de test en essayant tous les poids possibles. Nous constatons un gain moyen de 6%, pour un gain maximum de 15% sur le corpus de test, et sur le corpus d'apprentissage le gain moyen est de 9% pour un gain maximum de 17%.

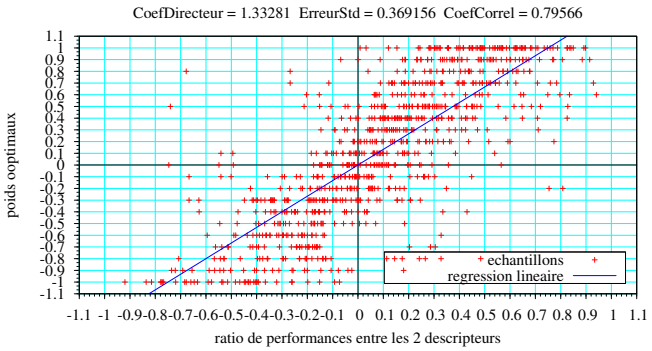


FIG. 5. Régression linéaire sur le corpus d'apprentissage

C. Sélection séquentielle des opérateurs

Les algorithmes de sélection de variables, ou sélection de caractéristiques, sont des techniques permettant d'extraire une information non redondante et pertinente de vecteurs de caractéristiques de grande dimension, typiquement les vecteurs produits par des descripteurs bas niveau dans le domaine de l'analyse d'images. Nous proposons d'adapter ce principe à la sélection de descripteurs dans la construction des réseaux d'opérateurs.

Nous souhaitons ainsi réduire le nombre d'opérateurs du réseau de façon à exclure les descripteurs non pertinents pour un concept donné, qui dans un schéma de fusion linéaire induiraient du bruit, et diminueraient donc la précision de détection du concept. En plus de la sélection de ces opérateurs notre adaptation de l'algorithme de sélection de caractéristiques vise également à déterminer des poids pour chacun des descripteurs retenus, de manière à renforcer l'importance des descripteurs de meilleure qualité dans la fusion finale.

Les algorithmes de sélection de caractéristiques peuvent être classés en deux grands groupes : les méthodes dites *filter* reposant

uniquement sur les propriétés intrinsèques des caractéristiques utilisées et celles appelées *wrapper* [36], qui au contraire définissent la pertinence des caractéristiques par l'intermédiaire d'une prédiction de la performance du système final.

Nous avons adopté un modèle *wrapper* afin d'exploiter l'information apportée par l'évaluation des différents ensembles d'opérateurs sur les données disponibles. Notre méthode consiste donc dans un premier temps à générer un sous-ensemble de caractéristiques (nous utilisons ici l'algorithme de sélection séquentielle avant, commencant par un ensemble vide et ajoutant successivement des variables), l'apprentissage est ensuite réalisé à partir de ces seules variables, puis l'on peut évaluer ces classifieurs par validation croisée, en sélectionnant enfin les sous-ensembles montrant les meilleures résultats de classification. Nous cessons la sélection séquentielle lorsque l'ajout d'un opérateur n'augmente plus les performances de classification du système.

En utilisant l'algorithme d'induction (le système de classification) comme une boite noire, cette approche peut être couteuse mais a montré de très bons résultats sur les benchmarks UC Irvine.

D. Descripteurs pondérés par leur performance

Pour fusionner les scores de prédiction issus de différents descripteurs une méthode utilisée dans différents articles consiste à pondérer chaque classifieur par ses performances individuelles calculée sur le corpus d'apprentissage par validation croisée. Ainsi les descripteurs pertinents pour la détection du concept se verront attribuer des poids plus importants, et auront donc plus d'influence dans le score de prédiction final résultant de la fusion.

Nous avons développé un modèle similaire en y incluant la fonction de prédiction de poids optimaux pour un couple de descripteurs, que nous avons obenue par régression, en l'étendant à un ensemble de n descripteurs. Celle-ci permettait d'estimer un poids W à partir des performances de deux descripteurs : $W = F(\frac{AP_2 - AP_1}{AP_1 + AP_2})$. Pour appliquer cette fonction F à un ensemble de n descripteurs nous proposons de calculer le poids de chacun de la façon suivante :

$$weight_i = \frac{1}{2} \times (1 - F(g(AP_i, \sum_{\substack{j=1 \\ j \neq i}}^{n} AP_j))$$

E. Optimisation globale des poids

La méthode précédemment proposée ne tient pas compte des relations qui peuvent exister entre descripteurs, ne déterminant le poids de chacun qu'à partir de ses performance individuelles. Ceci peut poser des problèmes si plusieurs opérateurs sont assez similaires. Notre dernier modèle propose donc de remédier à ce problème en optimisant globalement les poids des descripteurs, de manière à prendre en compte les relations de complémentarité ou de redondance qu'il peut exister entre eux.

IV. Expérimentations et résultats

A. Notre cadre d'expérimentation : TRECVID

TREC Video Retrieval est une campagne internationale d'évaluation proposée par le NIST[1] ayant pour but d'encourager la recherche en analyse de contenus vidéos, en fournissant pour cela une importante collection de test, des méthodes de mesure standards permettant de comparer les différents systèmes proposés, ainsi qu'un forum offrant aux différents participants la possibilité de présenter leur travaux.

TRECVID est souvent considéré comme le projet d'évaluation le plus complet de ces dernières années [37], nous avons donc décidé d'utiliser cette campagne comme cadre d'expérimentation de nos travaux afin de disposer d'un grand ensemble de données annotées utilisables pour l'apprentissage, et de pouvoir comparer sur un même corpus nos résultats à ceux de nombreuses équipes internationales actives dans ce domaine de recherche

Le corpus de vidéos proposé dans les campagnes TRECVID 2007 et 2008 est tiré de 400 heures de vidéos fournies par le Netherlands Institute For Sound And Vision, l'équivalent de l'INA en France. Concernant les données de 2008, la base d'apprentissage contient environ 100 heures de vidéos, découpées en 36262 plans extraits de 47 émissions de télévision différentes. La base de test (100 heures également) est constituée de 35766 plans provenant de 77 émissions. Les images-clés représentant chaque plan, extraites des flux vidéo, sont directement fournies aux participants sous forme d'images *JPG* de 352×288 pixels. Enfin, en plus des plans vidéos et des images clefs, TRECVID propose également la transcription de chaque séquence vidéo, résultant de la traduction automatique en anglais des phrases extraites de la bande son par des techniques de reconnaissance vocale (ASR[2]).

La campagne TRECVID comprend chaque année différentes tâches auxquelles les participants peuvent participer. Ces tâches sont évidement en rapport avec l'analyse de documents vidéos mais nécessitent des méthodes et des approches spécifiques. Nous nous intéressons ici à la tâche de détection de concept (*High Level Feature Extraction*), consistant à détecter un ensemble de concepts dans une base de vidéos. Afin de réaliser l'entraînement des systèmes, un corpus d'apprentissage est fourni, pour lequel on dispose d'annotations dans chacun des plans vidéos. Les concepts recherchés sont très variés, il sont tirés de l'ontologie LSCOM [38]. On peut les regrouper en différentes catégories : des objets (par exemple Car, Hand ou Bride), des scènes ou lieux (Beach, Outdoor, Classroom,..), des actions (Singing, Airplane-flying), des sujets thématiques (Military), ou encore des personnes (Bill-Clinton, Madeleine-Albright).

B. Plateforme logicielle

Afin de pouvoir mettre en oeuvre et évaluer les méthodes proposées dans ce rapport il nous a fallut implémenter une plateforme logicielle souple réalisant les différentes tâches d'entraînement, d'optimisation, de prédiction et d'évaluation. Cet ensemble de programmes s'est appuyé sur les travaux précédents de Stéphane Ayache, ancien doctorant de l'équipe MRIM, mais à nécessité d'importants remaniements. Ce framework a été développé en différents modules implémentés en *C*, *C++*, *Bash* et *Python* sur plateformes UNIX.

Les impératifs majeurs de la restructuration du code existant était la nécessité d'une plateforme générique et dynamique, autorisant une utilisation aisée de différents corpus d'apprentissage et de test, la possibilité de modifier la liste des concepts à détecter, et de spécialiser les réseaux d'opérateurs pour chacun, ainsi que de pouvoir facilement paramétrer les différents descripteurs utilisés. L'apprentissage et la reconnaissance sur des bases vidéos de la taille du corpus étant un processus coûteux en terme de calcul, l'accent à également été porté sur l'aspect distribué du programme, se déployant lors de l'exécution sur les différents serveurs de calcul mis a disposition par l'équipe MRIM.

[1] National Institute on Standads and Technology.

[2] Automatic Speech Recognition.

C. Description du système

La librairie *LIBSVM* [39] a été utilisé pour la mise en place des classifieurs SVM, en charge de l'apprentissage et de la prédiction. Parmi les différentes possibilités de noyaux SVM, le noyau RBF a été retenu, celui ci étant le plus couramment choisi dans le domaine de la classification d'image pour ses qualités optimales dans notre contexte d'application [40].

Le réglage des hyper-paramètres d'un classifieur SVM constitue une étape cruciale de la mise en place d'un système de classification efficace. Deux paramètres doivent être choisis : un paramètre relatif au noyau utilisé (γ dans le cas d'un noyau RBF), et le paramètre de régularisation (habituellement noté C), qui permet de jouer sur le compromis entre l'erreur sur la base d'apprentissage et la complexité du modèle. La recherche de paramètres adaptés est souvent appelée sélection de modèle dans la littérature, et il a été montré que ceux-ci influent fortement sur les performances du classifieur (se référer à [41]).

Pour cette sélection de modèle, nous avons opté pour une méthode classique de type "grid-search", consistant en une recherche systématique par discrétisation de l'espace des paramètres. Cette recherche fait de plus l'objet d'une validation croisée 5-fold. Il a certes été montré (notamment par [42]) que ces méthodes étaient coûteuses en calcul et n'offraient pas des performances optimales, mais cette solution est généralement adoptée pour des raisons de simplicité de mise en oeuvre. L'optimisation des paramètres de SVM permettrait probablement d'améliorer les performances générales de notre système, mais notre étude ne portait pas sur cet aspect du processus de classification. Parmi les pistes à éventuellement explorer dans le futur, généralement considérées meilleures à la méthode grid-search, [41] mentionne entre autre la stratégie de descente de gradients [43], les algorithmes génétiques [44], [45] ou l'utilisation de stratégies évolutionnaires [46].

Afin de pouvoir comparer nos résultats, nous avons réutilisé les descripteurs qu'employaient jusqu'alors Stéphane Ayache et Georges Quénot sur la plateforme précédente, soit 10 descripteurs dont par exemple des histogrammes de couleur, filtres de Gabor, Local Binary Patterns, ou encore des descripteurs SIFT.

La chaine de traitement peut être résumée de la façon suivante : l'extraction des images clés à partir des plans vidéos est réalisé par la segmentation en plans et sous-plans proposée par Fraunhofer HHI [47]. Chacun des descripteurs du réseau d'opérateur fourni en argument du programme est ensuite calculé pour l'ensemble des plans du corpus d'apprentissage, si cela n'avait pas été fait précédemment[1]. Les classifieurs SVM procèdent ensuite à l'apprentissage des modèles à partir de l'ensemble des exemples positifs du concept, et du double de plans négatifs tirés aléatoirement. La phase de prédiction consiste à extraire de la même manière les caractéristiques de chaque plan, pour la base de test cette fois ci, puis de prédire les scores de confiance du concept pour chacune des vidéos à partir des modèles SVM précédemment appris. Enfin il s'agit de fusionner ces différents scores selon le réseau décrit dans un fichier XML. Pour l'évaluation des résultats les vidéos du corpus de test sont classées des plus probables au moins probables pour le concept considéré, puis les 2000 premières sont comparées à la vérité terrain par le programme trec_eval. Les algorithmes que nous avons mis en place constituent une couche au dessus de ce noyau, qui vont générer des réseau, les évaluer sur différentes partitions de la base d'apprentissage, puis une fois achevée la construction de chacun des réseaux (spécifiques à chacun des concepts), on évaluera les performances finales sur le corpus de test.

[1] Les valeurs des descripteurs sont conservées sous forme de fichier.

D. Résultats

Comme nous l'avons expliqué dans la partie IV-A nous avons utilisé les données de la campagne TRECVID 2008 pour l'évaluation et la validations de nos modèles. La phase de construction des réseaux pour chacun des concepts a été réalisée sur le corpus d'apprentissage, en utilisant des méthodes de validation croisée. Puis ces réseaux ont été appliqués sur le corpus de test dans la phase de prédiction des concepts. Enfin nous avons pu mesurer les performances finale de notre système à partir des scores générés pour chacun des plans de la base de test, et de la vérité terrain mise à disposition par TREC.

Afin de mesurer l'apport de nos techniques nous avons comparé nos résultat au réseau ayant montré les meilleures performance dans les différentes expérimentations de Stéphane Ayache et Georges Quénot, qui consistait en une fusion tardive de l'ensemble des descripteurs, combinés sans pondération. Nous appellerons ce réseau de référence *"baseline"*. Plusieurs ensembles de descripteurs ont été testés. Nous avons d'abord conduit nos expérimentations avec l'intégralité des 10 descripteurs disponibles,, puis en ne retenant que 4 d'entre eux choisis pour leur faible corrélation (c'est à dire des descripteurs fournissant des informations de natures différentes), et enfin en n'en gardant que 3 (histogramme de couleurs, orientation de contours, et sift)/

Les résultats de ces expérimentations sont reportés dans le tableau I. On constate, lors de l'utilisation de l'intégralité de descripteurs un gain relatif de performance non significatif pour la pondération basée sur les performances, une baisse importante pour l'algorithme de sélection séquentielle, et une augmentation de 3% des performances pour l'optimisation globale des descripteurs. En revanche lorsque l'on réduit l'ensemble d'opérateurs à 3 ou 4 descripteurs la sélection séquentielle montre de bons résultats, la pondération AP un gain de 7.8% et l'optimisation globale de 6.3%.

Notons que les pourcentages reportés pour chacun des concepts correspondent aux gains relatifs de performance par rapport au réseau de référence. On observe une grande variation dans les valeurs observés, les différentes approches pouvant montrer de très forts gains pour certains concepts (jusqu'à 374% pour le concept Kitchen, passant d'une précision moyenne de 0.009 à 0.04) ou au contraire une chute de leurs performances. Cependant l'impact de ces gains sur le score MAP global dépend des performances absolues, qui montrent globalement des valeurs assez resserrées, variant pour la plupart entre -0.02 et +0.02.

Quelles conclusions pouvons nous tirer au vu de ces expérimentations ? Il est très difficile d'interpréter des résultats dans un domaine comme la recherche d'images ou de vidéos étant donné le nombre et la complexité des facteurs influant sur les scores de classification. On observe rarement un gain franc sur l'ensemble des concepts ou au contraire une démarche totalement néfaste, lorsque l'on regarde les différentes parutions publiées lors des conférences TRECVID, la plupart des nouveautés proposées par les différentes équipes montrent en général des gains discutables, présentant une amélioration sur certains concepts, mais une baisse sur d'autres, et on se réfère donc souvent au score MAP global.

Néanmoins nous avons pu tirer plusieurs constat des expérimentations. Premièrement que les relations de dépendance entre les descripteurs sont cruciales sur le choix de stratégie à adopter. La

TAB. I
RÉSULTATS SUR LE CORPUS 2008 POUR UN ENSEMBLE DE 3 DESCRIPTEURS

concept	baseline	sélection seq.		pondération AP		optimisation globale	
Classroom	0.0143	0.0075	-47.55 %	0.0332	**132.17 %**	0.0305	**113.29 %**
Bridge	0.0099	0.0053	-46.46 %	0.0104	**5.05 %**	0.0073	-26.26 %
Emergency-Vehicle	0.0071	0.0061	-14.08 %	0.0082	**15.49 %**	0.0083	**16.90 %**
Dog	0.2443	0.2376	-2.74 %	0.2355	-3.60 %	0.2389	-2.21 %
Kitchen	0.0089	0.0422	**374.16 %**	0.0085	-4.49 %	0.0088	-1.12 %
Airplane-flying	0.0391	0.0240	-38.62 %	0.1034	**164.45 %**	0.1017	**160.10 %**
Two-people	0.0526	0.0392	-25.48 %	0.0629	**19.58 %**	0.0621	**18.06 %**
Bus	0.0196	0.0038	-80.61 %	0.0152	-22.45 %	0.0111	-43.37 %
Driver	0.0725	0.1054	**45.38 %**	0.0628	-13.38 %	0.0529	-27.03 %
Cityscape	0.0937	0.1211	**29.24 %**	0.0962	**2.67 %**	0.0867	-7.47 %
Harbor	0.0162	0.0117	-27.78 %	0.0095	-41.36 %	0.0143	-11.73 %
Telephone	0.0066	0.0075	**13.64 %**	0.0055	-16.67 %	0.0053	-19.70 %
Street	0.1341	0.2369	**76.66 %**	0.1367	**1.94 %**	0.1343	**0.15 %**
Demonstration	0.0315	0.0461	**46.35 %**	0.0254	-19.37 %	0.0245	-22.22 %
Hand	0.0930	0.1398	**50.32 %**	0.1041	**11.94 %**	0.1050	**12.90 %**
Mountain	0.0421	0.0647	**53.68 %**	0.0445	**5.70 %**	0.0495	**17.58 %**
Nighttime	0.1086	0.1046	-3.68 %	0.1302	**19.89 %**	0.1183	**8.93 %**
Boat-Ship	0.0936	0.1274	**36.11 %**	0.0976	**4.27 %**	0.0984	**5.13 %**
Flower	0.1071	0.1033	-3.55 %	0.1077	**0.56 %**	0.1129	**5.42 %**
Singing	0.0284	0.0410	**44.37 %**	0.0201	-29.23 %	0.0298	**4.93 %**
MAP	0.0612	0.0738	**20.60 %**	0.0659	**7.72 %**	0.0650	**6.33 %**

pondération par AP ne tient pas compte de ces relations contrairement à l'approche d'optimisation globale, et nous constatons bien dans ce expérimentations que pour un ensemble de descripteurs "indépendants" l'approche globale se montre plus efficace, alors que pour des opérateurs présentant une redondance d'informations la pondération par AP lui est supérieure.

Lorsque beaucoup de descripteurs sont disponibles, la sélection incrémentale des opérateurs montre une forte tendance à exclure du réseau un grand nombre de ces descripteurs, jugés non nécessaires car n'augmentant pas les performances sur le corpus d'apprentissage. Ceux-ci peuvent cependant s'avérer utiles à la détection des concepts sur la base de test, si celle-ci présente des différences importantes avec la base d'apprentissage, et donc avoir de l'importance dans la construction d'un réseau plus généralisable.

1) Etude de l'influence du corpus sur les résultats: Nous avons lancer une dernière série d'expérimentation en entraînant les classifieurs sur le corpus d'apprentissage, puis en utilisant le corpus de test pour l'évaluation des performances dans la phase de construction des réseaux. Nous cherchions ainsi à voir, si, connaissant les performances relatives de chacun des descripteurs dans cette base de test, nos algorithmes permettraient ou non d'augmenter de manière importante les résultats de classification. Les résultats sont très concluants, avec un gain de près de 30%, et un MAP à 0.110. On voit donc qu'à partir de la même information initiale, c'est à dire les scores d'un ensemble de classifieurs, il est possible en les combinant de manière intelligente d'obtenir des résultats bien supérieurs à une fusion classique.

La seule information nécessaire pour parvenir à des résultats similaires serait une estimation relativement fiable des performances relatives de chaque descripteur. Nous pensions être en mesure d'obtenir une approximation de ces relations en évaluant par validation croisée leurs performances sur le corpus d'apprentissage, car on serait supposer observer des résultats relativement similaires d'un corpus à un autre, c'est d'ailleurs l'hypothèse sur laquelle repose la plupart des algorithme d'apprentissage supervisé (*IID*[1]). En effet si pour le concept 'Ciel' les descripteurs de couleurs fonctionnent mieux que les descripteurs de contours, cela traduit la nature du concept, et devrait donc être valable d'un corpus à un autre. Malheureusement nous avons constaté que les collections d'images fournies par TREC introduisent un très fort biais, les images-clés fournies n'étant souvent pas réellement représentatives des concepts, il devient très difficile d'estimer des paramètres généralisables au corpus de test.

Nous expliquons donc les résultats plutôt faibles de nos algorithmes par ces particularités liées au corpus TRECVID, empêchant la généralisabilité des modèles d'apprentissage. Signalons néanmoins, à titre d'indication, que notre meilleur run se positionnerait à la 40ème place parmi les 200 proposés dans la campagne 2008, gagnant ainsi 6 place par rapport au meilleur run de l'équipe. Ces biais liés à la collection de test que nous avons utilisé sont développés plus amplement dans la section suivante. Ils serait donc intéressant d'évaluer nos algorithmes sur une collection différente, comme celle de la campagne VOC, afin d'analyser leurs résultats, qui nous le pensons devraient être plus probants. Nous proposons cependant dans la section V plusieurs pistes qui permettraient de pallier à ces difficultés rencontrées sur des corpus comme celui de TRECVID.

2) Spécificités du corpus TRECVID: On constate d'une manière générale des résultats relativement faibles des différents systèmes proposés dans les campagnes TRECVID. Pour l'édition 2008 la précision moyenne de l'ensemble des participants étant de 0.0429. A titre de comparaison, sur la compétition VOC 2007[2], qui consistait à détecter une vingtaine de concepts dans des corpus d'images de taille similaire à TRECVID, la moyenne était égale à 0.4557, soit plus de dix fois supérieure. Cette différence était même encore plus significative sur l'édition 2006, avec un MAP moyen à 0.8627 pour VOC [48]. Cet important contraste ne peut s'expliquer par la qualité des systèmes proposés, qui adoptent globalement les mêmes approches et outils dans les différentes campagnes. Beaucoup d'équipes participent

[1] IID désigne l'hypothèse que des variables, dans notre cas les exemples d'un concept, soient indépendantes et identiquement distribuées.

[2] Visual Object Challenge.

d'ailleurs à ces deux challenges avec les mêmes systèmes. Alors d'où provient cette différence de performances ?

Une des raisons expliquant la difficulté d'apprentissage des concepts dans les campagnes TRECVID, et donc des résultats relativement faibles, provient de la politique d'annotation des images adoptée par TREC. Alors que la plupart des autres corpus d'apprentissages fournissent pour chaque concept des exemples représentatif de la classe à prédire, TREC a pris le parti d'annoter toutes les images où un utilisateur humain serait en mesure d'identifier le concept, y comprit s'il est partiellement occulté, ou s'il ne correspond qu'à une petite partie de l'image. Il en résulte une grande proportion d'images peu représentatives du concept à apprendre et donc difficilement exploitables par des algorithmes d'apprentissage artificiel. Les organisateurs revendiquent d'ailleurs ce choix dans [49], reconnaissant les difficultés que cela soulève pour un apprentissage performant mais souhaitant que les collections qu'ils proposent restent utilisables à long terme, lorsque les systèmes de reconnaissance auront probablement beaucoup évolué :

Face à ce problème, pour limiter l'impact négatif d'un trop grand nombre d'images non pertinentes dans l'apprentissage des concepts, plusieurs équipes ont même restreint le nombre d'images utilisées, en ne conservant que les plus représentatives de chaque concept, ceci en examinant manuellement toutes les images annotées du corpus. Cette démarche témoigne une fois de plus des problèmes rencontrés par les différents participants face aux spécificités du corpus fourni par TRECVID. Pour comparer ces différences nous présentons dans la figure 6 quelques images issus de la base d'apprentissage de TREC, ainsi que de celle de VOC, pour le concept "Airplane", proposé dans les deux.

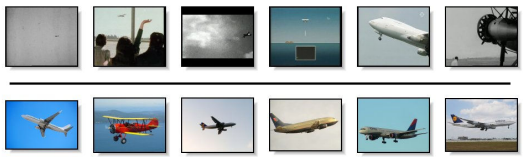


FIG. 6. Images annotées pour le concept "Airplane" dans les campagnes TREC (en haut) et VOC (en bas)

Un autre biais ayant un fort impact sur les résultats obtenus est le choix des concepts. Beaucoup d'autres campagnes d'évaluation se focalisent sur la recherche de concepts "concrets", c'est à dire des objets tangibles. Les organisateurs de TRECVID ont en revanche choisi des concepts plus proches des requêtes que pourrait avoir un utilisateur réel, notamment s'il recherche des personnes (concepts "Bill-Clinton" ou "Madeleine-Albright"), ou bien des activités, ou des évènements ("Protest/Demonstration", "Singing", "Running",...). Or il est clair que ces concepts sont bien plus difficiles à détecter de manière automatique, car trop éloignés des informations visuelles. Comme le souligne Yang *et al.* il est en effet dur d'imaginer pouvoir différencier à partir des couleurs et textures des concepts comme "Crowd" et "Protest", ou "Prisoner" et "Corporate-Learder", qui requièrent une fine analyse, et donc des processus bien plus complexes que l'amélioration des techniques proches du signal. Il suggère donc que les efforts se concentrent sur des concepts moins "sémantiques", de niveau intermédiaire, ayant par conséquent une plus forte cohérence avec une analyse basée sur les caractéristiques bas-niveau [50].

V. PERSPECTIVES

Comme nous l'avons expliqué dans l'analyse des résultats, la généralisabilité des modèles appris sur le corpus d'apprentissage est la condition nécessaire à de bonnes performances. Les différentes méthodes de construction automatique de réseaux que nous avons proposées s'appuient en effet sur l'estimation par validation croisée des performances relatives de chaque opérateur. Une grande variation de ces mesures d'un corpus à un autre aura donc un impact négatif sur le choix du réseau optimal. Nous présentons maintenant plusieurs pistes qui permettraient de compenser cette différence de distribution entre corpus, ou de l'intégrer au processus de création des réseaux.

A. Estimation de la généralisabilité

L'équipe Informedia a proposé dans [51] une méthode intéressante d'estimation de la généralisabilité d'un modèle d'apprentissage, définie par le déclin relatif entre les performances AP_{test} mesurées sur le corpus d'évaluation, et celles sur la base de test (AP_{test}) :

$$Déclin\_relatif = \frac{AP_{test} - AP_{dev}}{AP_{test}}$$

Leurs études ont montré que ce déclin était fortement corrélé aux méta-propriétés des modèles SVM, notamment le ratio de données positives, le ratio de vecteurs de support, et la distribution des prédictions du modèle sur le corpus de test (cette distribution est caractérisée par deux variables : *score_range*, correspondant à l'écart entre la plus faible et la plus haute prédiction du modèle sur l'ensemble des données de test, et *max_score* étant le plus haut score prédit). Ils ont donc construit par régression[1] un modèle de prédiction du déclin entre AP_{dev} et AP_{test}, à partir des méta-propriétés citées précédemment.

En utilisant cette estimation $\widehat{AP}_{test}$ dans notre sélection de réseaux d'opérateurs nous pourrions ainsi sélectionner les réseaux pour lesquels l'estimation des performances sur le corpus de test serait la plus haute, au lieu des plus efficaces sur le corpus de développement.

B. Apprentissage semi-supervisé

Une autre voie possible permettant d'enrichir les méthodes que nous avons présenté dans ce mémoire serait l'utilisation de techniques d'apprentissage semi-supervisé. Comme nous l'avons montré dans l'analyse des résultats, les différentes pistes de combinaison automatique de descripteurs que nous avons exploré sont basées sur leurs performances, que l'on a calculées sur des sous ensembles du corpus d'apprentissage en mesurant la précision des prédictions produites. Or le principal problème à la généralisation de ces méthodes provient de l'importante variation dans la distribution et la nature des exemples entre la base d'apprentissage et la base de test, cependant pour obtenir une meilleure stabilité des prédiction entre le corpus d'apprentissage et le corpus de test, une solution serait d'intégrer des données de la base de test à nos modèles grâce à des algorithmes d'apprentissage semi-supervisé, c'est à dire à exploiter les données non-annotées de la base de test en plus de la collection d'apprentissage [52].

Ces approches sont généralement adoptées lorsque le volume d'exemples étiquetés disponible est trop faible, l'annotation manuelle par des humains étant bien plus coûteuse que la collecte de documents non étiquetés. Notre motivation est différente, nous disposons en effet d'un nombre suffisant d'exemples annotés pour chacune des classes à détecter, mais nous souhaitons obtenir des modèles plus proches de

[1] En utilisant l'algorithme SVR (Support Vector Regression).

la distribution du corpus de test, pour une meilleure généralisabilité des réseaux d'opérateurs construits.

Parmi les différentes types d'apprentissage supervisés, dont une étude plus approfondie est proposée dans "Semi-supervised learning literature survey" [52], nous retiendrons les approches de Self-Training et Co-Training [53], permettant d'incorporer incrémentalement à l'algorithme d'apprentissage des exemples de la base de test.

Une deuxième piste interessante pour prendre en compte les données de la base de test dans les modèles d'apprentissage est l'apprentissage transductif détaillé par Vapnik dans [54]. L'idée principale, telle que présentée dans [55], consiste à prédire les annotations du corpus de test qui optimisent la frontière de décision séparant les exemples positifs et négatifs sur le corpus d'apprentissage et le corpus de test. Cela revient à trouver une prédiction des classes sur les instance de la base de test maximisant les marges du classifieur SVM.

C. Contexte

Terminons enfin ce par d'autres perspectives plus larges qui ont retenu notre attention au cours de l'exploration de ce sujet. Les voies les plus prometteuses dans la détection de concepts nous paraissent être la prise en compte du contexte. Ce contexte peut prendre différentes formes, dont nous mentionnons certaines ici.

1) Informations spatiales: Comme le souligne [56], les informations relatives aux relations spatiales entre régions devraient contribuer globalement à l'amélioration des performances, par exemple pour les concepts "Sky" et "Airplane_flying", ou encore pour "Sea" et "Boat_ship".

Une façon de prendre en compte cette information spatiale est proposée par [57]. A partir de descripteurs locaux SIFT l'image est décrite comme un sac de mots visuels[1] (ensembles de points d'intérêt). La représentation classique en mots visuels ignore l'information spatiale. Pour intégrer cet aspect l'image est divisé en régions rectangulaires de tailles égales, puis les mots visuels sont calculés pour chacune des régions, et finalement concaténés en un vecteur global. Un ensemble d'expérimentations a montré qu'un partitionnement 2x2 (image divisée en 2 colonnes et 2 lignes) obtenait de meilleurs résultats que sur l'image globale

2) Clustering: Lorsque les exemples appartenant à une même classe, utilisés pour l'entraînement d'un classifieur, sont de nature et distribution trop hétérogènes l'algorithme d'apprentissage montre souvent des difficultés à modéliser et caractériser la classe de façon efficace. Dans cette situation il serait utile de diviser ces exemples en groupes homogènes par des méthodes de clustering, puis de diviser la classe à apprendre en différentes sous-classes correspondant à chacun des clusters. Des classifieurs différents seraient donc entraînés sur ces groupes d'exemples, puis lors de la prédiction, à partir des scores de chacune des sous-classes et de systèmes de fusion par vote on pourrait déterminer le score de la classe mère.

Cette approche permettrait de combiner l'apprentissage supervisé avec une séparation préalable des données par clustering (apprentissage non supervisé) et ainsi de prendre en compte le *contexte image* des concepts, c'est à dire la nature des images dans lequel il apparaît. L'expérimentation de ce procédé sur différents domaines de test (voir [58]) a montré des résultats très prometteurs qu'il serait intéressant d'appliquer à la reconnaissance de concepts dans des vidéos.

3) Corrélations entre concepts: Une des pistes les plus intéressantes pour une amélioration des systèmes d'indexation est selon nous l'utilisation des relations qu'il peut exister entre concepts. Un concept apparaît en effet rarement seul dans une image, si l'on recherche une voiture il y'a de fortes chances que le concept "route" soit également présent dans le plan, or la majorité des systèmes adoptent une classification séparée de chacun des concepts, sans exploiter ces relations. Cependant de plus en plus de méthodes tirant parti de ces informations commencent à voir le jour.

Une première approche simple proposée dans [59] consiste à regrouper manuellement des concepts au sein de groupes sémantiques, décrivant des contextes d'occurrence similaires. En entraînant d'une part un classifieur pour chaque concept et d'autre part un clasifieur global pour chaque catégorie ils ont ensuite fusionné ces résultats lors de la détection d'un concept donné pour un plan du corpus de test en multipliant le score du classifieur spécifique au concept par celui du groupe auquel il appartient. Bien que simple cette méthode à néanmoins montré de bon résultats, avec un MAP de 0.1381, 5% supérieur à leur *baseline*.

Partant de l'hypothèse que les concepts ne sont pas indépendants, l'équipe CMU avait introduit en 2006 un autre modèle exploitant les relations entre plusieurs concepts sémantiques dans les vidéos [60]. Leur approche utilise une fusion multi-concept nommée MDRF (*Multiple Discriminative Random Field*), à partir de laquelle ils construisent un graphe non-orienté représentant ces relations sémantiques. En utilisant de nombreux concepts intermédiaires, tirés d'autres collections de données annotées, on peut de plus augmenter le support inter-concepts. Plusieurs universités ont mis à disposition leurs détecteurs de concept, parmi lesquelles Mediamill-101 [61], Columbia374[62] et Vireo374.[63].

Les concepts de ces trois collections de détecteurs sont tirés de l'ontologie LSCOM. L'utilisation de la structure hiérarchique de l'ontologie ou d'autres éléments, comme la distance sémantique entre concepts que l'on pourrai obtenir de lexiques tels que Wordnet [64] permettrai à notre sens une gestion plus fine que la seule analyse des co-occurrences de concepts. Nous n'avons trouvé aucun article mentionnant ce type d'approches qui nous sembleraient pourtant pertinentes, les schéma de fusion pourriant être adaptés en fonction de la nature des relations entre les concepts (synonymes, inclusion, antonymes,...), nature qu'il est très difficile d'obtenir en observant simplement les co-occurrences.

D. Conclusion

Nous avons présenté dans ce mémoire une étude des grandes approches et tendances actuelles dans le domaine de la détection de concepts dans des documents multimédia en livrant une analyse des pistes explorées ces dix dernières années et de leurs résultats. Nous avons ainsi tenté de proposer au lecteur une vision la plus large possible de ce domaine de recherche actif en dressant le bilan des problématiques considérées comme "résolues" par la communauté CBIR[2], qui servent maintenant de base aux nouvelles approches prometteuses s'attaquant aux verrou majeur du franchissement du fossé sémantique.

Face aux limites de l'approche par réseaux d'opérateurs proposée dans [35] nous avons également développé un ensemble d'algorithmes adressant le problème de la construction automatique de ces réseaux. Ces modèles intègrent notamment une fusion adaptative des

[1] Bag-Of-Visual-Words.

[2] Content Based Image Retrieval.

données, en vue d'une organisation intelligente de la combinaison des différentes sources d'informations, en détectant de manière automatique les plus pertinentes dans le processus de classification, et ceci spécifiquement à chaque concept .

Ces travaux ont demandé le développement d'une plateforme logicielle pérenne, modulaire, qui offre la flexibilité nécessaire à la conduite d'expérimentations sur différents corpus, listes de concepts, ou ensembles de descripteurs. Celle ci pourra servir d'outil aux recherches futures de l'équipe, constituant une alternative aux implémentations ad-hoc généralement effectuées pour les besoin d'un cadre d'application précis. Une fois cette base logicielle mise en place, nous avons ensuite pu y intégrer les modèles de fusion que nous avons proposés et évaluer leurs résultats, qui sont présentés et interprétés dans la section IV-D.

Ces modèles semblent prometteurs au vu des premières expérimentations malgrés le fort biais induit par les particularités du corpus TRECVID que nous avions expliquées lors de l'analyse des résultats. Le trop grand nombre d'images non représentatives des concepts qu'elles sont pourtant censées illustrer fausse ainsi l'estimation de paramètres généralisables. Ce problème auquel est confronté la plupart des participants à cette campagne conduit à des résultats relativement faibles (la moyenne des scores des différentes équipes est près de dix fois inférieure à ceux relevés sur la campagne VOC). Il serait donc intéressant d'une part d'évaluer nos méthodes sur des corpus plus stables, et d'autre part de gérer cette grande variation entre corpus par des méthodes de bootstrap, d'estimation du déclin relatif ou d'apprentissage semi-supervisé. Ces différentes pistes ont été détaillées dans les perspectives, et constituent des voies qui nous semblent pertinentes pour la suite de ces travaux.

Nous avons enfin livré une réflexion plus ouverte sur l'avenir de la recherche de documents multimédias, et la poursuite de nos travaux. Nous pensons notamment que pour être à même de détecter des concepts haut-niveau de manière efficace les approches centrées sur l'amélioration des descripteurs visuels, ou l'optimisation des classifieurs ne suffisent pas. Il est nécessaire d'adopter conjointement des solutions permettant de tirer parti de l'information fournie par le reste de l'image ou de la vidéo pour intégrer une notion de contexte aux prédictions. Ce contexte doit pouvoir exploiter les relations de co-occurrence entre concepts dans le processus de classification, en recherchant par exemple plusieurs concepts corrélés au sein de l'image. De cette manière l'identification d'une voiture pourrait venir renforcer la prédiction du concept "route". D'autre types de relations peuvent également être inclues dans la notion de contexte, comme les relations spatiales ou temporelles.

De grands défis restent donc à relever pour parvenir à une indexation performante des documents multimédia, mais les efforts sont de plus en plus nombreux dans cette direction. Alors que la recherche textuelle est maintenant considéré comme mûre, avec notamment la généralisation des moteurs de recherche de pages web, celle des vidéos reste un domaine plus jeune mais en plein essor. L'analyse d'image par le contenu trouve de plus un grand nombre d'applications autres que la recherche de documents, par exemple pour la robotique, la vidéo-surveillance, l'informatique ambiante, la reconnaissance biométrique, ou encore l'imagerie médicale.

RÉFÉRENCES

[1] R. C. Veltkamp and M. Tanase, "A survey of content-based image retrieval systems," *Content-based image and video retrieval*, pp. 47–101, 2002. I

[2] A. Hampapur, T. E. Weymouth, and R. Jain, "Feature based digital video indexing," in *Proceedings of the third IFIP WG2. 6 working conference on Visual database systems 3 (VDB-3) table of contents*. Chapman & Hall, Ltd. London, UK, UK, 1997, pp. 115–141. 1

[3] J. Eakins, M. Graham, and U. of Northumbria at Newcastle, *Content-based image retrieval*. University of Northumbria at Newcastle, 1999. I-B

[4] B. A. Draper, J. Bins, and K. Baek, "ADORE adaptive object recognition," *Lecture notes in computer science*, pp. 522–537, 1998. I-C

[5] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluation of color descriptors for object and scene recognition," in *IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008*, 2008, pp. 1–8. I-C, II-B1, III

[6] M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer, "Learning object representations for visual object class recognition," 2007. I-C

[7] E. Yilmaz and J. A. Aslam, "Estimating average precision with incomplete and imperfect judgments," in *Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM New York, NY, USA, 2006, pp. 102–111. II-A

[8] W. Forstner, *A framework for low level feature extraction*. Springer-Verlag, 1994. II-B

[9] M. J. Swain and D. H. Ballard, "Indexing via color histograms," in *Computer Vision, 1990. Proceedings, Third International Conference on*, 1990, pp. 390–393. II-B1

[10] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Transactions on circuits and systems for video technology*, vol. 11, no. 6, pp. 703–715, 2001. II-B1

[11] M. Stricker and M. Orengo, "Similarity of color images," in *Proc. SPIE Storage and Retrieval for Image and Video Databases*, vol. 2420. San Jose CA USA, 1995, pp. 381–392. II-B1

[12] F. Mindru, T. Tuytelaars, L. V. Gool, and T. Moons, "Moment invariants for recognition under changing viewpoint and illumination," *Computer Vision and Image Understanding*, vol. 94, no. 1-3, pp. 3–27, 2004. II-B1

[13] J. R. Smith and S. F. Chang, "Tools and techniques for color image retrieval," *Storage & Retrieval for Image and Video Databases IV*, vol. 2670, p. 426–437, 1996. II-B1

[14] G. Pass and R. Zabih, "Histogram refinement for content-based image retrieval," in *Applications of Computer Vision, 1996. WACV'96., Proceedings 3rd IEEE Workshop on*, 1996, pp. 96–102. II-B1

[15] J. Huang, S. R. Kumar, M. Mitra, and W. J. Zhu, *Image indexing using color correlograms*. Google Patents, 2001. II-B1

[16] Y. Deng, B. S. Manjunath, C. Kenney, M. S. Moore, and H. Shin, "An efficient color representation for image retrieval," *IEEE Transactions on Image Processing*, vol. 10, no. 1, pp. 140–147, 2001. II-B1

[17] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *IEEE Transactions on systems, man and cybernetics*, vol. 3, no. 6, pp. 610–621, 1973. II-B1

[18] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp. 55–73, 1990. II-B1

[19] T. Ojala, M. Pietikainen, and T. Maenpaa, "Gray scale and rotation invariant texture classification with local binary patterns," *Lecture Notes in Computer Science*, vol. 1842, pp. 404–420, 2000. II-B1

[20] G. R. Cross and A. K. Jain, "Markov random field texture models," in *Conference on Pattern Recognition and Image Processing, Dallas, TX*, 1981, pp. 597–602. II-B1

[21] A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 661–674, 1984. II-B1

[22] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *International Journal of Computer Vision*, vol. 60, no. 1, pp. 63–86, Oct. 2004. [Online]. Available : http://dx.doi.org/10.1023/B:VISI.0000027790.02288.f2 II-B3

[23] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," *International Journal of Computer Vision*, vol. 73, no. 3, pp. 263–284, 2007. [Online]. Available : http://dx.doi.org/10.1007/s11263-006-9967-1 II-B3

[24] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, no. 10, pp. 1615–1630, 2005. II-B3

[25] D. G. Lowe, "Distinctive image features from Scale-Invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004. [Online]. Available : http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94 II-B3

[26] Y. Ke and R. Sukthankar, "PCA-SIFT : a more distinctive representation for local image descriptors," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2, 2004, pp. II–506–II–513 Vol.2. II-B3

[27] W. Freeman and E. Adelson, "The design and use of steerable filters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 13, no. 9, pp. 891–906, 1991. II-B3

[28] C. Schmid and R. Mohr, "Local grayvalue invariants for image retrieval," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 19, no. 5, pp. 530–535, 1997. II-B3

[29] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 4, pp. 509–522, 2002. II-B3

[30] V. N. Vapnik, *The nature of statistical learning theory*. springer, 1995. II-C1, II-C1

[31] I. Bloch, *Fusion d'informations en traitement du signal et des images*, ser. Traité IC2, série Traitement du signal et de l'image. Lavoisier 2000-2009, 2003. II-D1

[32] C. G. M. Snoek, M. Worring, and A. W. M. Smeulders, "Early versus late fusion in semantic video analysis," in *Proceedings of the 13th annual ACM international conference on Multimedia*. Hilton, Singapore : ACM, 2005, pp. 399–402. [Online]. Available : http://portal.acm.org/citation.cfm?id=1101149.1101236# II-D2

[33] S. Ayache and G. Quenot, "LIG and LIRIS at TRECVID 2008 : High level feature extraction and collaborative annotation," in *Proc. of TRECVID Workshop*, 2008, pp. 17–18. 2

[34] S. Ayache, "Indexation de documents vidéos par concepts par fusion de caractéristiques audio, vidéo et texte," Ph.D. dissertation, INPG Grenoble, 2007. II-E

[35] S. Ayache and G. Quénot, "Indexation de documents multimédia par réseaux d'opérateurs," *Coria 2007*, p. 385, 2007. 3, II-E, III, V-D

[36] G. H. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in *Proceedings of the Eleventh International Conference on Machine Learning*, vol. 129. New Brunswick, NJ, USA, Morgan Kaufmann, 1994. III-C

[37] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval : State of the art and challenges," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP)*, vol. 2, no. 1, pp. 1–19, 2006. IV-A

[38] M. Naphade, J. R. Smith, J. Tesic, S. F. Chang, W. Hsu, L. Kennedy, A. Hauptmann, and J. Curtis, "LSCOM large-scale concept ontology for multimedia," *IEEE MULTIMEDIA*, pp. 86–91, 2006. IV-A

[39] C. Lin and C. Chang, "LIBSVM a library for support vector machines," *Software available at http ://www.csie.ntu.edu.tw/~cjlin/libsvm*, 2001. IV-C

[40] C. Snoek, v. d. K.E.A. Sande, d. O. Rooij, B. Huurnink, v. J.C. Gemert, J. Uijlings, J. He, X. Li, I. Everts, V. Nedovic, v. M. Liempt, v. R. Balen, F. Yan, M. Tahir, K. Mikolajczyk, J. Kittler, d. M. Rijke, J. Geusebroek, T. Gevers, M. Worring, A. Smeulders, and D. Koelma, "The MediaMill TRECVID 2008 semantic video search engine," in *TRECVID 2008 : Proceedings of the 2008 TREC Video Retrieval Evaluation workshop*, 2008, pp. 1–14. [Online]. Available : http://dare.uva.nl/record/301644 IV-C

[41] C. Chatelin, S. Adam, Y. Lecourtier, L. Heutte, T. Paquet, and Y. Oufella, "Optimisation multi-objectif pour la sélection de modèles SVM," *AFCET/AFRIF-AFIA RFIA - Reconnaissance de Formes et Intelligence Artificielle*, vol. 8, 2008. IV-C

[42] S. M. LaValle, M. S. Branicky, and S. R. Lindemann, "On the relationship between classical grid search and probabilistic roadmaps," *Algorithmic Foundations of Robotics V*, p. 59, 2003. IV-C

[43] K. M. Chung, W. C. Kao, C. L. Sun, L. L. Wang, and C. J. Lin, "Radius margin bounds for support vector machines with the RBF kernel," *Neural Computation*, vol. 15, no. 11, pp. 2643–2681, 2003. IV-C

[44] C. L. Huang and C. J. Wang, "A GA-based feature selection and parameters optimizationfor support vector machines," *Expert systems with applications*, vol. 31, no. 2, pp. 231–240, 2006. IV-C

[45] C. H. Wu, G. H. Tzeng, Y. J. Goo, and W. C. Fang, "A real-valued genetic algorithm to optimize the parameters of support vector machine for predicting bankruptcy," *Expert Systems with Applications*, vol. 32, no. 2, pp. 397–408, 2007. IV-C

[46] F. Friedrichs and C. Igel, "Evolutionary tuning of multiple SVM parameters," *Neurocomputing*, vol. 64, pp. 107–117, 2005. IV-C

[47] C. Petersohn, "Fraunhofer HHI at TRECVID 2004 : Shot boundary detection system," in *TREC Video Retrieval Evaluation Online Proceedings, TRECVID*, 2004. IV-C

[48] M. Everingham, A. Zisserman, C. Williams, L. V. Gool, and K. U. Leuven, "The pascal visual object classes challenge 2006 (voc2006) results," in *Workshop in ECCV06, May. Graz, Austria*, 2006. IV-D2

[49] A. F. Smeaton, P. Over, and W. Kraaij, "High-Level feature detection from video in TRECVid : a 5-Year retrospective of achievements," *Multimedia Content Analysis : Theory and Applications—Divakaran A., ed*, 2008. IV-D2

[50] J. Yang and A. Hauptmann, "Reliability of video concept detection," in *Proc. Int. Conf. Image and Video Retrieval*, 2008, p. 85–94. IV-D2

[51] M. G. Christel, A. G. Hauptmann, W. H. Lin, M. Y. Chen, J. Yang, L. Mummert, R. V. Baron, S. Schlosser, X. Sun, and V. Valdes, "Informedia @ TRECVID2008 exploring new frontiers," *TRECVID 2008 : Proceedings of the 2008 TREC Video Retrieval Evaluation workshop*, 2008. V-A

[52] X. Zhu, "Semi-supervised learning literature survey," *Computer Science, University of Wisconsin-Madison*, 2006. V-B

[53] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*. Madison, Wisconsin, United States : ACM, 1998, pp. 92–100. [Online]. Available : http://portal.acm.org/citation.cfm?id=279962 V-B

[54] V. Vapnik, "Statistical learning theory. 1998," *NY Wiley*, 1998. V-B

[55] T. Joachims, "Transductive inference for text classification using support vector machines," in *Sixteenth International Conference on Machine Learning*, 1999. V-B

[56] O. Kucuktunc, M. Bastan, U. Gudukbay, and O. Ulusoy, "BILKENT UNIVERSITY MULTIMEDIA DATABASE GROUP AT TRECVID 2008," in *TREC Video Retrieval Evaluation Proceedings*, 2008. V-C1

[57] S. F. Chang, J. He, Y. G. Jiang, E. E. Khoury, C. W. Ngo, A. Yanagawa, and E. Zavesky, "Columbia University/VIREO-CityU/IRIT TRECVID2008 High-Level feature extraction and interactive video search," in *TRECVID workshop*, 2008. V-C1

[58] N. Japkowicz, "Supervised learning with unsupervised output separation," in *International Conference on Artificial Intelligence and Soft Computing*, 2002, pp. 321–325. V-C2

[59] Y. Peng, Z. Yang, J. Yi, L. Cao, H. Li, and J. Yao, "Peking university at TRECVID 2008 : High level feature extraction," in *TREC Video Retrieval Evaluation Workshop*, 2008. V-C3

[60] A. G. Hauptmann, M. Y. Chen, M. Christel, W. H. Lin, R. Yan, and J. Yang, "Multi-lingual broadcast news retrieval," in *NIST TRECVID Workshop, Gaithersburg, MD*, 2006. V-C3

[61] C. G. M. Snoek, M. Worring, J. C. V. Gemert, J. M. Geusebroek, and A. W. M. Smeulders, "The challenge problem for automated detection of 101 semantic concepts in multimedia," in *Proceedings of the 14th annual ACM international conference on Multimedia*. ACM New York, NY, USA, 2006, pp. 421–430. V-C3

[62] A. Yanagawa, S. F. Chang, L. Kennedy, and W. Hsu, "Columbia university's baseline detectors for 374 LSCOM semantic visual concepts," *Columbia University ADVENT technical report*, vol. 222, pp. 2006–8, 2007. V-C3

[63] Y. G. Jiang, C. W. Ngo, and J. Yang, *VIREO-374 LSCOM semantic concept detectors using local keypoint features*. Citeseer, 2007. V-C3

[64] C. Fellbaum, *WordNet An electronic lexical database*. MIT press Cambridge, MA, 1998. V-C3