HAL open science

Utilisation du contexte pour l'indexation sémantique des images et vidéos

Abdelkader Hamadi

To cite this version:

Abdelkader Hamadi. Utilisation du contexte pour l'indexation sémantique des images et vidéos. Intelligence artificielle [cs.AI]. Université de Grenoble, 2014. Français. NNT : 2014GRENM047 . tel-01551793

HAL Id: tel-01551793

https://theses.hal.science/tel-01551793v1

Submitted on 30 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE GRENOBLE

Thèse

Pour obtenir le grade de

Docteur de l'Université de Grenoble

Spécialité : **Informatique**

Arrêté ministériel : 7 Août 2006

Présentée par

Abdelkader Hamadi

Thèse dirigée par **Georges Quénot**
et codirigée par **Philippe Mulhem**

préparée au sein **du Laboratiore d'informatique de Grenoble**
et de **l'École Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique (MSTII)**

Utilisation du contexte pour l'indexation sémantique des images et vidéos

Thèse soutenue publiquement le **23 Octobre 2014**,
devant le jury composé de :

Mr. Liming Chen
Professeur, l'École centrale de Lyon, LIRIS, Président
M. Philippe-Henri Gosselin
Professeur, ENSEA, ETIS, Université de Cergy, France, Rapporteur
M. Hervé Glotin
Professeur, Université du Sud Toulon-Var, Rapporteur
Mr. Alexandre Benoit
Maître de conférence, LISTIC- Polytech'Savoie, Examinateur
M. Georges Quénot
Directeur de recherche CNRS, CNRS, (Membre), Directeur de thèse
M. Philippe Mulhem
Chargé de recherche CNRS, CNRS, (Membre), Co-Directeur de thèse

Résumé

L'indexation automatisée des documents image fixe et vidéo est un problème difficile en raison de la "distance" existant entre les tableaux de nombres codant ces documents et les concepts avec lesquels on souhaite les annoter (personnes, lieux, événements ou objets, par exemple). Des méthodes existent pour cela mais leurs résultats sont loin d'être satisfaisants en termes de généralité et de précision. Elles utilisent en général un ensemble unique de tels exemples et le considère d'une manière uniforme. Ceci n'est pas optimal car un même concept peut apparaître dans des contextes très divers et son apparence peut être très différente en fonction de ces contextes. Dans le cadre de cette thèse, nous avons considéré l'utilisation du contexte pour l'indexation des documents multimédia. Le contexte a largement été utilisé dans l'état de l'art pour traiter diverses problématiques. Dans notre travail, nous retenons les relations entre les concepts comme source de contexte sémantique. Pour le cas des vidéos, nous exploitons le contexte temporel qui modélise les relations entre les plans d'une même vidéo. Nous proposons plusieurs approches utilisant les deux types de contexte ainsi que leur combinaison, dans différents niveaux d'un système d'indexation. Nous présentons également le problème de détection simultanée de groupes de concepts que nous jugeons lié à la problématique de l'utilisation du contexte. Nous considérons que la détection d'un groupe de concepts revient à détecter un ou plusieurs concepts formant le groupe dans un contexte ou les autres sont présents. Nous avons étudié et comparé pour cela deux catégories d'approches. Toutes nos propositions sont génériques et peuvent être appliquées à n'importe quel système pour la détection de n'importe quel concept. Nous avons évalué nos contributions sur les collections de données TRECVid et VOC, qui sont des standards internationaux et reconnues par la communauté. Nous avons obtenu de bons résultats, comparables à ceux des meilleurs systèmes d'indexation évalués ces dernières années dans les compagnes d'évaluation précédemment citées.

Abstract

The automated indexing of image and video is a difficult problem because of the“distance” between the arrays of numbers encoding these documents and the concepts (e.g. people, places, events or objects) with which we wish to annotate them. Methods exist for this but their results are far from satisfactory in terms of generality and accuracy. Existing methods typically use a single set of such examples and consider it as uniform. This is not optimal because the same concept may appear in various contexts and its appearance may be very different depending upon these contexts. In this thesis, we considered the use of context for indexing multimedia documents. The context has been widely used in the state of the art to treat various problems. In our work, we use relationships between concepts as a source of semantic context. For the case of videos, we exploit the temporal context that models relationships between the shots of the same video. We propose several approaches using both types of context and their combination, in different levels of an indexing system. We also present the problem of multiple concept detection. We assume that it is related to the context use problematic. We consider that detecting simultaneously a set of concepts is equivalent to detecting one or more concepts forming the group in a context where the others are present. To do that, we studied and compared two types of approaches. All our proposals are generic and can be applied to any system for the detection of any concept. We evaluated our contributions on TRECVID and VOC collections, which are of international standards and recognized by the community. We achieved good results comparable to those of the best indexing systems evaluated in recent years in the evaluation campaigns cited previously.

Remerciements

Merci à ALLAH avant tout, pour tout, et en tous les cas.

Je présente ma sincère gratitude à mes encadreurs, Mr. Georges Quénot et Mr. Philippe Mulhem qui m'ont dirigé, conseillé et répondu présents tout le long de ma thèse, tout en m'offrant assez de liberté de travail, sans oublier leurs qualités humaines et leur compréhension surtout dans les moments difficiles par rapport à ma maladie. Je leur dit également merci pour l'environnement de travail sympathique et professionnel qu'ils m'ont procuré dans lequel j'ai appris tant de choses. Merci encore une fois.

Je remercie les membres du juty, Pr. Hervé Glotin et Pr. Philippe-Henri Gosselin d'avoir accepté d'être les rapporteurs de mon mémoire et aussi pour leurs remarques et suggestions qui m'ont permis d'avoir de nouvelles perspectives à mon travail. Je remercie également les examinateurs Pr. Liming Chen et Mr. Alexandre Benoit.

Je tiens à remercier les membres du groupe MRIM pour leur sympathie et pour le temps que nous avons passé ensemble tout le long de ma thèse. Merci parce que j'ai trouvé auprès de vous, une deuxième famille dans une période ou j'étais et je ne pouvais tenir seul. Un merci spécial à Catherine Berrut pour sa disponibilité et ses précieux conseils surtout les jours précédant ma soutenance qui m'ont aidé à présenter un travail de meilleure qualité.

Je ne peux oublier de remercier les partenaires du projet MRIM GDR-ISIS pour leur collaboration pour réaliser ce travail. Cette thèse a été financé par OSEO dans le cadre du projet Quaero. Une part des expérimentations a été réalisée sur la plateforme GRID5000 qui a été développée par INRIA ALADDIN avec l'aide de CNRS RENATER, plusieurs universités ainsi que d'autres organismes de financements. Je présente mes remerciements à l'ensemble de ces structures.

La suite de mes remerciements est destinée à ma famille, mon entourage et mes amis.

Merci à mes parents, ALLAH seul sait pourquoi ; sans eux ce chapitre de ma vie n'aurait pu être écrit. Merci à eux pour tout ce qu'ils ont fait pour moi et pour ce qu'ils ont fait de moi. Je tiens aussi à leur avouer que je n'ai voulu être Docteur que pour leur rendre hommage et honneur et que j'ai réalisé ce travail d'une part grâce à leurs prières et bienveillance. J'ai attendu 29 ans pour leur dire "Merci infiniment" de la manière dont ils méritent. J'espère que cela puisse vous faire plaisir.

Merci à mes deux frères et ma grande sœur et mes adorables neveux.

Merci à mon tonton Ahmed ainsi que son épouse Louisa pour leur générosité et tout ce qu'ils ont fait pour moi. Je remercie mes tantes sans exception surtout ma tante Fatma qui m'a toujours épaulé et soutenu. Merci à tous mes cousins et cousines sans exception.

Un merci spécial à Nadia, Mohannad et Bahjat. Nadia, nous avons commencé la thèse le même jour, nous avons traversé le même chemin et nous avons partagé à peu près les mêmes difficultés et les mêmes joies durant presque quatre années, je ne peux te dire que merci pour l'amie que tu as été. Mohannad, je n'oublierai jamais ta présence, toujours là pour me dépanner, merci pour ta simplicité, ta générosité et ta présence. Bahjat, je te dis merci pour ta présence et ton soutien, aussi bien scientifiquement que moralement. Tu as a toujours été présent via tes qualités humaines, surtout les derniers jours de ma thèse, je ne te remercierai jamais assez.

Merci à Chahro, Abdelwahab, Roukhou, Amine, Ismail, Fouad Baabaa, Fifi et Hind d'avoir marqué cette période spéciale de ma vie, pas la peine que j'en dise plus parce que vous êtes pour moi des êtres spéciaux.

Merci à tous ceux qui m'ont aidé de près ou de loin à terminer ce chapitre important de ma vie. Sans citer des noms, chaque personne se reconnaitra. Merci à mes amis sans exception. Merci aussi à ceux que j'ai connus dans les dernières pages, qui ont su donner un goût sympa à ma vie dans une période compliquée. Merci à ceux qui m'ont supporté et ont supporté mon agressivité, mes réactions brusques dues à la pression et à la charge à laquelle j'étais soumis durant une longue période. Merci à ceux qui ont accepté et ont su s'adapter à mon statut de fantôme.

Merci à ceux que je n'ai pas cités et qui pourtant le méritent. Je suis sûr qu'ils se reconnaîtront et c'est le principal.

Les deux derniers remerciements sont spéciaux et vont à deux personnes spéciales : Lyliche et ma future épouse.

Merci encore une autre fois à ALLAH pour tout, avant et après tout.

Table des matières

Table des matières

1 Introduction générale **1**
- 1.1 L'indexation sémantique de documents multimédia 1
- 1.2 Difficultés et défis de l'indexation sémantique 3
  - 1.2.1 Les fossés sémantique et sensoriel 3
  - 1.2.2 Problème des classes déséquilibrées 4
- 1.3 Contexte et problématique 6
- 1.4 Objectif de la thèse 8
- 1.5 Contributions / Plan de la thèse 10

2 État de l'art **13**
- 2.1 Documents image et vidéo 13
- 2.2 Notion de concept 15
- 2.3 Indexation par le contenu des documents multimédia 17
  - 2.3.1 Description d'un système d'indexation multimédia 20
- 2.4 Description de documents image et vidéo 22
  - 2.4.1 Descripteurs locaux et descripteurs globaux 23
  - 2.4.2 Descripteurs visuels 23
    - 2.4.2.1 Descripteurs de couleur 23
    - 2.4.2.2 Descripteurs de texture 24
    - 2.4.2.3 Descripteurs de formes 25
    - 2.4.2.4 Descripteurs basés sur les points d'intérêts 25
  - 2.4.3 Descripteurs audio 26
  - 2.4.4 Descripteurs de mouvements 27
  - 2.4.5 Discussion 28
- 2.5 Agrégation des descripteurs locaux 29
  - 2.5.1 Sacs de mots visuels 29
  - 2.5.2 Noyaux de Fisher 29
- 2.6 Optimisation des descripteurs 30
  - 2.6.1 Normalisation des descripteurs 31
    - 2.6.1.1 Transformation de loi de puissance (Power-law) . 32
  - 2.6.2 Réduction de dimensionnalité 32
    - 2.6.2.1 L'analyse en composantes principales (ACP) . . . 32
    - 2.6.2.2 L'analyse en composantes indépendantes (ACI) . 33
    - 2.6.2.3 L'analyse discriminante linéaire (LDA) 33

2.7 Apprentissage automatique/Classification 34
2.7.1 Approches supervisées vs. Non supervisées 35
2.7.2 Approches génératives 36
2.7.3 Approches discriminatives 36
2.7.3.1 K-plus proches voisins 37
2.7.3.2 Approches à noyaux 37
2.7.4 Apprentissage par modèle d'ensemble 39
2.7.4.1 Voting 40
2.7.4.2 Bagging 40
2.7.4.3 Boosting 41
2.7.4.4 Stacking 42
2.7.5 Apprentissage profond (Deep learning) 42
2.7.6 Normalisation des sorties de classificateurs 43
2.8 Fusion 46
2.8.1 Fusion précoce 46
2.8.2 Fusion tardive 47
2.8.3 Fusion de noyaux 48
2.9 Ré-ordonnancement 49
2.10 Les ontologies 50
2.11 Utilisation du contexte pour l'indexation sémantique 51
2.11.1 Définition du contexte 52
2.11.2 Pourquoi avoir besoin du contexte ? 53
2.11.3 Catégorisation du contexte 55
2.11.3.1 Choix des types de contexte retenus 63
2.11.4 Comment utiliser le contexte ? 64
2.12 Détection simultanée de plusieurs concepts 66
2.13 Évaluation 67
2.13.1 Évaluation des systèmes d'indexation 67
2.13.2 Campagnes d'évaluation 68
2.13.2.1 Image clef 68
2.13.2.2 Pascal-VOC 69
2.13.2.3 TRECVid 69
2.14 Conclusion 71

3 Contributions pour l'utilisation du contexte sémantique 72
3.1 Notations 73
3.2 Ré-ordonnancement sémantique basé sur une ontologie 73
3.2.1 Description de l'approche 74
3.2.2 Expérimentations et résultats 76
3.3 Reclassement sémantique par regroupement 79
3.3.1 Description de l'approche 80
3.3.2 Expérimentations et résultats 82
3.4 Rétroaction conceptuelle 85
3.4.1 Description de l'approche 86
3.4.2 Expérimentations et résultats 89
3.5 Rétroaction conceptuelle étendue 92

3.5.1 1 : Pondération des dimensions conceptuelles 92
3.5.2 2 : Filtrage de concepts à base de relations sémantiques . . 94
3.5.3 Expérimentations et résultats 95
3.6 Rétroaction conceptuelle itérative 98
3.6.1 Description de l'approche 98
3.6.2 Expérimentations et résultats 99
3.7 Conclusion 101

4 Contributions pour l'utilisation du contexte temporel 103
4.1 Re-scoring temporel 103
4.1.1 Description de l'approche 103
4.1.2 Expérimentations et résultats 105
4.2 Descripteurs temporels 107
4.2.1 Description de l'approche 108
4.2.2 Expérimentation et résultats 110
4.3 Conclusion 112

5 Contributions pour l'utilisation conjointe des contextes sémantique et temporel 114
5.1 Re-scoring à deux couches 114
5.1.1 Description de l'approche 114
5.1.2 Expérimentations et résultats 115
5.2 Rétroaction conceptuo-temporelle 117
5.2.1 Description de l'approche 117
5.2.2 Expérimentations et résultats 118
5.3 Conclusion 123

6 Contribution pour la détection simultanée de groupes de concepts dans les documents multimédia 124
6.1 Description des approches 124
6.1.1 Modèles de concepts multiples 124
6.1.2 Fusion de détecteurs de concepts individuels 126
6.2 Expérimentations et résultats 127
6.2.1 TRECVid 127
6.2.2 VOC 130
6.3 Conclusion 134

7 Conclusion et perspectives 135
7.1 Conclusion 135
7.1.1 Contributions dans cette thèse 136
7.1.2 Points forts de la thèse et limitations 139
7.2 Perspectives 140

A Annexe A 143
A.1 Le concepts TRECVid évalués 143
A.2 Les concepts TRECVid utilisés (ceux évalués et non) 143
A.3 Les concepts de ImageCLEF 145

B Annexe B **147**
B.1 Descripteurs de video(Genérés par les partenaires IRIM) 147

C Annexe C **150**
C.1 Liste de publications 150

Bibliographie **174**

Chapitre 1

Introduction générale

1.1 L'indexation sémantique de documents multimédia

Avec l'explosion de la masse de données multimédia, le développement d'applications et de moteurs de recherche pour l'exploiter devient crucial. Dans le cadre de ce travail, un document multimédia est toute unité qui peut être retrouvée par le système. Un tel document est pour nous un échantillon multimédia quelconque : une image, une vidéo complète, un plan vidéo, etc. Pour être retrouvé, un document doit être indexé, c'est-à-dire caractérisé par le système, et c'est sur cet aspect que porte ce travail de thèse.

Historiquement, le premier type de méthode proposé propose des "recherches par l'exemple" : l'utilisateur fournit un document exprimant son besoin, qui est comparé à l'ensemble des documents de la base de données, une liste de documents triée en terme de ressemblance par rapport à la requête est alors fournie par le système en réponse. Concrètement, ce ne sont pas les documents bruts qui sont comparés, mais des représentations, ou "descripteurs". Ces descripteurs sont souvent "de bas niveau", c'est-à-dire qu'ils caractérisent des aspects proches du signal brut. Les comparaisons entre ces descripteurs reposent sur des mesures de distance. La figure 1.1 décrit cette approche.

Au cours du temps, la recherche par l'exemple a atteint ses limites à cause de l'impossibilité d'exprimer tous les besoins via des exemples (avec une photographie d'un bâtiment, exprime-t-on que l'on recherche ce bâtiment ou un bâtiment similaire ?). Les utilisateurs, habitués à rechercher des documents par les moteurs de recherche sur le web pas exemple, voient le texte comme le moyen le plus simple pour exprimer leur besoin d'information. Cette solution, simple pour l'utilisateur, est très compliquée à réaliser. Cette difficulté ne se résume pas en la substitution de la modalité dans laquelle la requête est exprimée, mais elle ouvre la possibilité d'exprimer des requêtes sémantiques complexes, auxquelles il s'avère très difficile de répondre via les méthodes existantes dans l'état de l'art. De plus, la difficulté majeure liée à ce cas réside dans la tâche de correspondance entre les termes textuels composant la requête et des données brutes (valeurs numériques codant les images/vidéos).

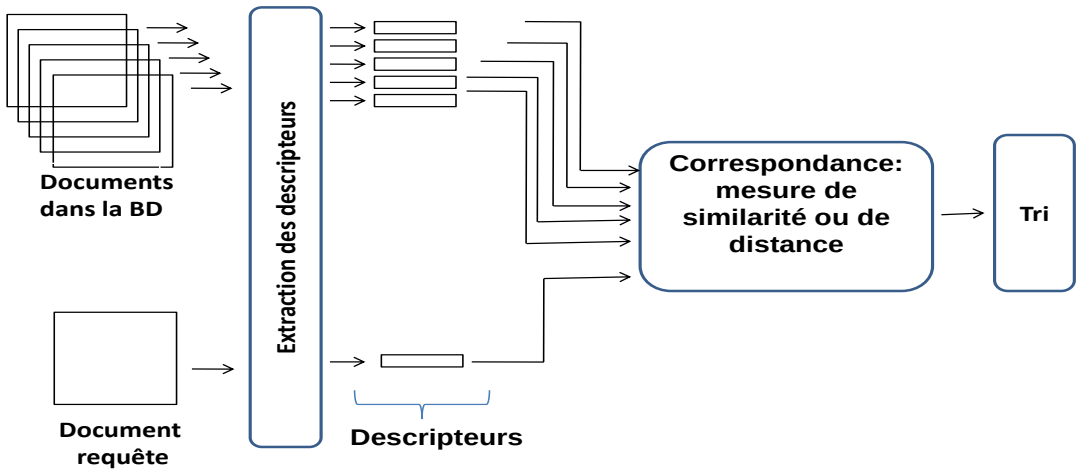


FIGURE 1.1 – Recherche par l'exemple.

Des solutions liées à la description des documents multimédia par des termes textuels ont été apportées par les communautés de l'apprentissage automatique et du multimédia. Les méthodes proposées reposent sur l'apprentissage automatique supervisé : pour rechercher les images contenant un "cheval" par exemple, ces méthodes utilisent un ensemble d'images d'apprentissage contenant un "cheval" (exemples positifs) et d'autres images ne contenant pas cet objet (les exemples négatifs). Dans ce contexte, "cheval" est appelé : le concept cible. Nous verrons plus loin plus de détails sur cette notion. À partir de ces exemples d'entraînement, les solutions existantes apprennent un modèle permettant de classifier de façon binaire les descripteurs de bas niveau extraits d'une image échantillon, en décidant si une image contient ou non le concept cible. La figure 1.2 décrit brièvement ce type d'approches.

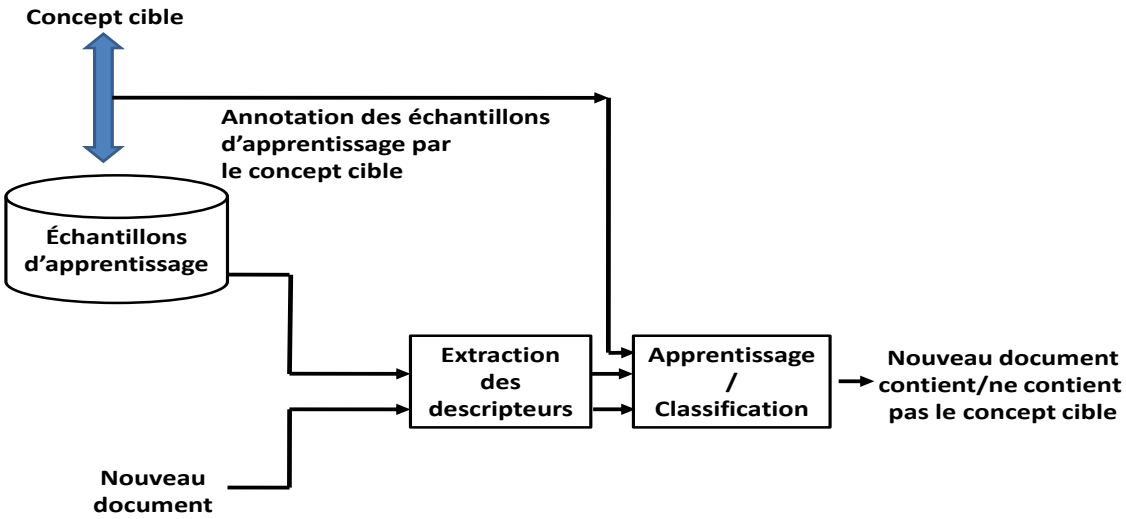


FIGURE 1.2 – Recherche par apprentissage supervisé.

Spécifier qu'un échantillon est positif ou négatif par rapport à un concept donné est appelé annotation automatique. Les méthodes basées sur l'apprentissage sont généralement appliquées sur un ensemble des concepts cibles considérés indépendamment les uns des autres via une classification binaire. Les exemples d'apprentissage sont généralement annotés manuellement. Plus l'ensemble de données annotées est grand, plus les méthodes d'apprentissage supervisées sont efficaces. Or, avec l'explosion de la masse de données multimédia, il devient de plus en plus difficile, voire impossible d'annoter manuellement de gros corpus de données d'images et vidéos, en raison du coût très élevé nécessaire en termes de temps et argent. Le passage à une annotation automatique [1] s'est avéré donc impératif. Dans ce travail, nous utilisons aussi l'expression "détection de concepts" pour désigner l'indexation de documents par des concepts.

De manière plus générale, l'annotation automatique des documents image fixe et vidéo est une tâche difficile à cause de plusieurs raisons, parmi lesquelles : le fossé sensoriel, le fossé sémantique, et le problème de classes déséquilibrées lié aux méthodes d'apprentissage supervisé. Nous détaillons dans la section suivante ces problèmes qui constituent les défis majeurs de l'indexation multimédia.

1.2 Difficultés et défis de l'indexation sémantique

1.2.1 Les fossés sémantique et sensoriel

Devant faire correspondre une représentation brute (de bas niveau) à une description conceptuelle ou sémantique, l'indexation automatique se heurte à un problème appelé "le fossé sémantique" (*semantic gap* en anglais). Selon Ayache [Aya07] : "*Le fossé sémantique est ce qui sépare les représentations brutes (tableaux de nombres) et sémantiques (concepts et relations) d'un document numérique*". Smeulders et al. [SWS+00], quant à eux, donnent une autre définition : "*Le fossé sémantique est le manque de concordance entre les informations que la machine peut extraire d'un document numérique et des interprétations humaines*". Les mêmes auteurs évoquent un autre type de fossé dit "fossé sensoriel", qui est défini comme le fossé existant entre le monde réel 3D et sa représentation en une image 2D. Lors de l'acquisition des images et vidéos, cette projection vers un espace 2D provoque une grande perte d'informations. Cela mène à une représentation de bas niveau, pas très efficace. Les caractéristiques visuelles de bas niveau ne parviennent souvent pas à décrire les concepts sémantiques de haut niveau [ZH00]. D'autre part, la prise en photo ou en vidéo d'une même scène ou même objet par des dispositifs différents, dans des situations différentes, mène à des documents multimédia différents en termes de :

- Luminosité/illumination (figure 1.3(a)) : La présence d'ombre ou de halos sur un objet provoque parfois des occultations partielles ;
- Taille de l'objet ou l'échelle de l'image : (figure 1.3(b)) ;

1. Dans l'état de l'art, on utilise le terme indexation ou annotation.

- L'angle de prise de vue (figure 1.3(c)) : un même objet pris de différents angles de vues peut apparaitre sous des formes variées, parfois n'aidant pas à savoir qu'il s'agit du même objet ;

Le fossé sémantique est également accentué à cause de :

- La variabilité visuelle intra-classe (figure 1.3(d)) : les objets d'une même classe n'ont pas toujours les même caractéristiques visuelles, on parle de "variations visuelles" ou de "multiples représentations d'un même objet" ;
- La variabilité visuelle inter-classes (figure 1.3(e)) : des descriptions visuelles similaires peuvent concerner deux concepts qui n'ont rien à voir l'un à l'autre (des descripteur visuels similaires concernant deux objets différents) ;

Tous ces problèmes compliquent la tâche à la machine pour déduire que ces contenus numériques qu'elle manipule correspondent au concept recherché ou non.

Franchir le fossé sémantique constitue une des difficultés majeures d'un système automatique d'annotation/indexation de documents multimédia. Les communautés de la vision par ordinateur et de l'indexation automatique continuent de traiter ce problème. Beaucoup de travaux ont été réalisés dans le but d'augmenter la corrélation entre des contenus visuels similaires sémantiquement en proposant de bons descripteurs. D'autres méthodes d'apprentissage automatique ont été proposées, qui permettent de faire correspondre plus efficacement les caractéristiques de bas niveau à des descriptions sémantiques ou des concepts.

1.2.2 Problème des classes déséquilibrées

Un des défis de l'indexation et de la recherche multimédia est la nécessité de passage à l'échelle : très grandes collections et un grand nombre de concepts cibles. Cela engendre un des problèmes majeurs des méthodes d'apprentissage supervisées : les corpus de données deviennent hautement déséquilibrés. En effet, il est très fréquent qu'une large majorité d'exemples d'apprentissage soient annotés par rapport à une seule des deux classes. La performance des algorithmes standards d'apprentissage et de classification est sensiblement affectée par ce problème, parce qu'ils sont souvent basés sur l'optimisation de la précision ou du taux d'erreurs. Cette optimisation est biaisée par la classe majoritaire. D'autre part, parce que le modèle sera généré à base d'un ensemble insuffisant d'exemples de la classe minoritaire, il aura sans doute un très faible pouvoir de généralisation, parce qu'il a appris à classer les exemples positifs sur trop peu d'exemples de la classe minoritaire [FTTUG10]. Le tableau 1.1 présente la distribution des exemples positifs, négatifs ainsi que leur ratio, pour certains concepts dans la collection TRECVid 2012 [OAM+12]. On constate qu'il y a peu de concepts qui ont plus d'échantillons positifs que négatifs, et ce genre de concepts sont souvent génériques, et ils obtiennent donc leur fréquence élevée via l'occurrence des concepts qui leur sont spécifiques, comme par exemple le concept "Personne" qui est un concept père de plusieurs autres concepts fils : Masculin, Féminin, Enfant, Garçon, Fille, Acteur, etc. Une occurrence de l'un de ces concepts fils implique nécessairement l'occurrence de leur concept père. De ce fait, le concept père aura plus d'exemples positifs

(a) Changement de luminosité

(b) Changement d'échelle (taille de l'objet), exemple de vélo

(c) Changement de l'angle de prise de vue

(d) Différence entre les caractéristiques visuelles d'un même objet (exemple de la classe *avion*)

(e) Ambiguïté visuelle. Un même contenu visuel ou deux contenus visuels similaires peut référer à deux sens différents

FIGURE 1.3 – Quelques difficultés auxquelles se heurte l'indexation automatique des documents multimédia. Les images sont tirées de la thèse de R.Benmokhtar ([Ben09]).

que chacun de ses concepts fils. Pour la majorité des concepts on a une centaine d'exemples positifs *vs.* des centaines de milliers d'exemples négatifs.

Afin de traiter ce problème, plusieurs approches ont été proposées dans l'état de l'art. Une d'entre elles consiste à pré-traiter le corpus d'apprentissage de façon à minimiser les différences entre les classes. En d'autres termes, ces méthodes d'échantillonnage modifient la répartition des classes minoritaire et majoritaire dans l'ensemble de données de façon à obtenir un nombre plus équilibrée d'instances dans chaque classe. Beaucoup de travaux se sont basés sur le sur-échantillonnage ou le sous-échantillonnage de la classe majoritaire/minoritaire.

concepts	# négatifs	# positifs	ratio
Personne	21452	71287	1 :3
Acteur	52801	3292	16 :1
Chat	86813	231	376 :1
Bus	76521	135	567 :1
Soleil	42686	180	237 :1
Basketball	94774	151	628 :1
Football	85925	196	438 :1

TABLE 1.1 – Fréquences de certains concepts dans la collection TRECVid 2012.

D'autres chercheurs se sont focalisés sur 1) l'utilisation de sources d'informations externes, notamment les ontologies, 2) l'utilisation d'outils d'apprentissage et la proposition d'approches modifiées pour associer les descripteurs de bas niveau aux concepts sémantiques, etc.

1.3 Contexte et problématique

Malgré la rapide augmentation du nombre de publications de travaux dans le domaine de la recherche d'information basée sur le contenu (CBIR) [Dow93, ZZ95], les problèmes liés au fossé sémantique ne sont pas encore résolus. Utiliser des descripteurs de bas niveau via des algorithmes sophistiqués ne peut pas modéliser d'une manière efficace la sémantique de l'image/vidéo. En effet, cette approche a de nombreuses limites surtout lorsqu'il s'agit de larges corpus de données [MR01], car il n'existe aucun lien direct entre le niveau signal et le niveau sémantique [SCS01, IMA09].

Plusieurs efforts ont visé à réduire le fossé sémantique, et à traiter ou contourner certains autres problèmes rencontrés dans le domaine de l'indexation multimédia. Ces efforts ont été concrétisés par une amélioration significative de la performance des systèmes d'indexation d'images et/ou de vidéos par détection de concepts. Par exemple, dans la compagne TRECVid [SOK06], la performance du meilleur système varie entre 0.2 et 0.3 sur une échelle allant de 0 à 1, selon la métrique officielle d'évaluation de cette compagne (précision moyenne).

Même si la performance des systèmes d'indexation des images et vidéos s'est améliorée ces dernières années, cette amélioration ne concerne pas tous les concepts. La figure 1.4 montre les performances obtenues pour certains concepts par le système d'indexation *Quaero* qui a décroché une place dans le top 5 de la compagne d'évaluation TRECVid 2012. On peut remarquer qu'il y a des concepts qu'on peut détecter efficacement. D'autres part, la performance pour certains autres est moyenne, alors qu'on signale un échec des systèmes actuels d'indexation pour la détection de certains concepts donnant lieu à des résultats médiocres.

En général, les méthodes d'indexation traitent les concepts indépendamment les uns des autres. Ceci n'est pas optimal car un même concept peut apparaître dans des contextes très divers et son apparence peut être très différente en fonc-

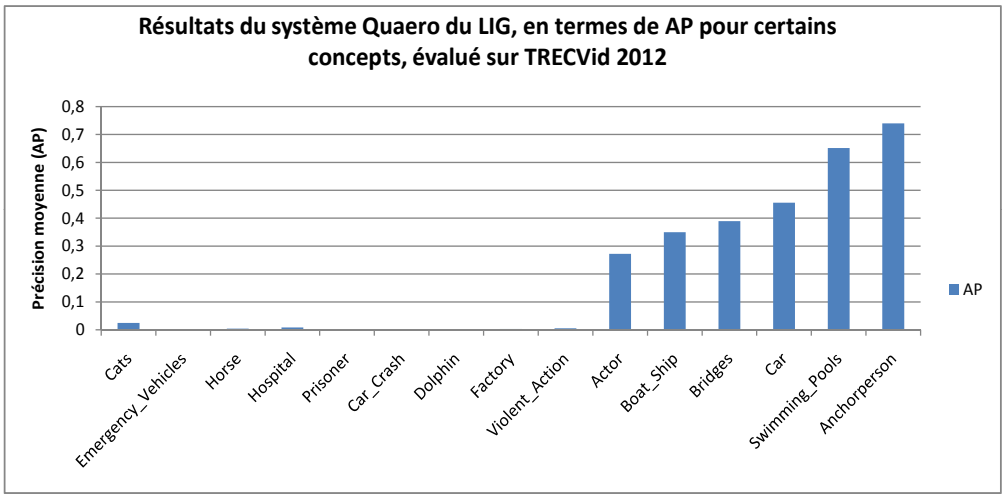


FIGURE 1.4 – Performance du système Quaero pour la détection de certains concepts.

tion de ces contextes. Le contexte peut être par exemple : le type d'émission (journal télévisé, fiction, divertissement, publicité, etc.), la date, le lieu, le pays ou la culture de diffusion ou de production, ou encore les modalités présentes ou absentes (cas de documents en noir et blanc et/ou sans son par exemple). Le contexte peut également être considéré comme un autre concept ou un ensemble d'autres concepts. L'utilisation des relations inter-concepts peut être soutenue par le fait qu'une image est très riche en sémantique, et que souvent, un concept n'apparait pas seul dans une image ou une vidéo. Une étude des co-occurrences des concepts dans les corpus TRECVid nous a confirmé cette remarque.

Nous concluons que tenter de chercher un concept tout seul dans une image ou une vidéo n'est pas la meilleure approche. Il est important de souligner qu'un humain se sert dans son raisonnement de certaines sémantiques et liens entre les concepts pour déduire l'occurrence de certains concepts dans une image ou une vidéo. En effet, l'humain n'a pas besoin de chercher la forme d'un objet pour déduire sa présence, il peut se servir pour cela de l'occurrence de certains autres sémantiques (e.g. L'occurrence d'un lit exclut l'apparition d'un avion).

Une caractéristique qui différencie les vidéos des images fixes est l'aspect temporel. Cette source d'information peut s'avérer cruciale pour les systèmes d'indexation de vidéos. En effet, certains concepts occurrent d'une façon continue dans des plans successifs. Cette remarque concerne exclusivement les concepts qui font référence à des évènements (e.g. Explosion, accident, mariage, anniversaire, etc) ou même certains autres concepts qui représentent une dynamique et non des objets ou des choses fixes, comme par exemple : Football, Basketball, Atterrissage, etc. Quand ce genre de concepts apparaissent dans une vidéo, leur apparition se propage sur plus d'un seul plan et couvre donc un ensemble de plans successifs. Nous appelons cette source d'information *le contexte temporel*.

Les contextes sémantique (relations inter-concepts) et temporel représentent une source d'information très importante à ne pas négliger, qui s'avère cruciale pour réduire encore plus le fossé sémantique, et donc, améliorer les systèmes de détection de concepts dans les images/vidéos.

1.4 Objectif de la thèse

Cette thèse s'inscrit dans le cadre de l'indexation sémantique des images et vidéos par détection de concepts. Le but principal est de proposer des approches permettant d'exploiter des informations contextuelles pour améliorer la performance d'un système d'indexation de documents multimédia. Pour réaliser cet objectif il est nécessaire avant tout de développer un système d'indexation des images et vidéos. D'autre part, la performance du système à améliorer est un détail important. En effet, parce qu'il est plus difficile d'améliorer un système qui est déjà bon, nous avons tenu à appliquer nos contributions sur un système de référence ayant une performance raisonnablement bonne. Pour ce faire, nous avons utilisé un système d'indexation d'images et de vidéos, développé au sein de l'équipe MRIM [2] du laboratoire d'informatique de Grenoble (LIG). L'effort est ensuite investi pour incorporer des informations contextuelles dans ce système dans le but d'améliorer ses performance en termes de précision.

Avant d'exploiter les informations contextuelles, il faudrait tout d'abord définir ce que c'est que le contexte. En effet, le contexte n'a pas une définition précise, et a été utilisé dans l'état de l'art dans différents domaines de différentes façons, et dans chaque travail de recherche les informations contextuelles sont de différentes natures et issues de différentes sources. Il est donc primordial de se situer par rapport aux différentes définitions possibles du contexte et de fournir une définition qui correspond à notre problématique de recherche. Après avoir défini le contexte, le travail de cette thèse s'articule autour des questions saillantes suivantes :

1. Quelles sont les sources du contexte ?
2. Comment utiliser les informations contextuelles ?
3. A quel niveau du système d'indexation les informations contextuelles seront elles incorporées ?

Au lieu de considérer les concepts et les échantillons multimédia d'une manière indépendante, nous proposons d'utiliser les relations inter-concepts pour le contexte sémantique. Pour le contexte temporel, nous allons considérer les relations entre les segments d'un même document doté d'un aspect temporel, comme décrit dans la figure 1.5. Le défi est d'atteindre des performances meilleures que celles obtenues par les systèmes de référence.

Nous étudions ensuite, différentes approches possibles pour exploiter le contexte dans différentes étapes du système d'indexation, comme décrit dans la figure 1.6. Le but de cette étude est de déduire la configuration la mieux adaptée pour une utilisation optimale du contexte.

2. Modélisation et Recherche d'Information Multimédia

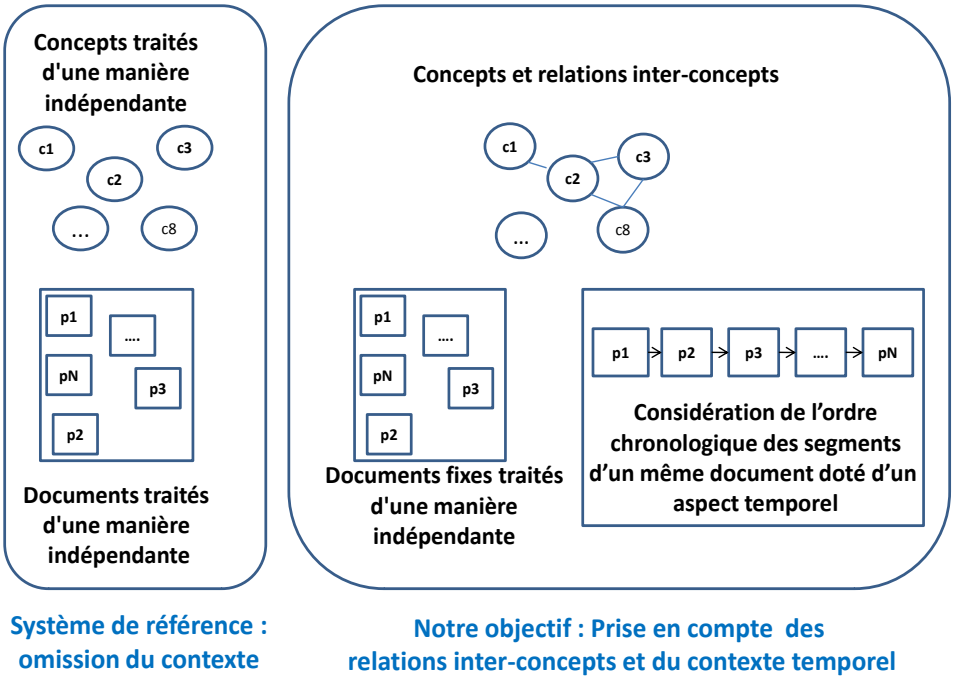


FIGURE 1.5 – Positionnement de notre travail de thèse.

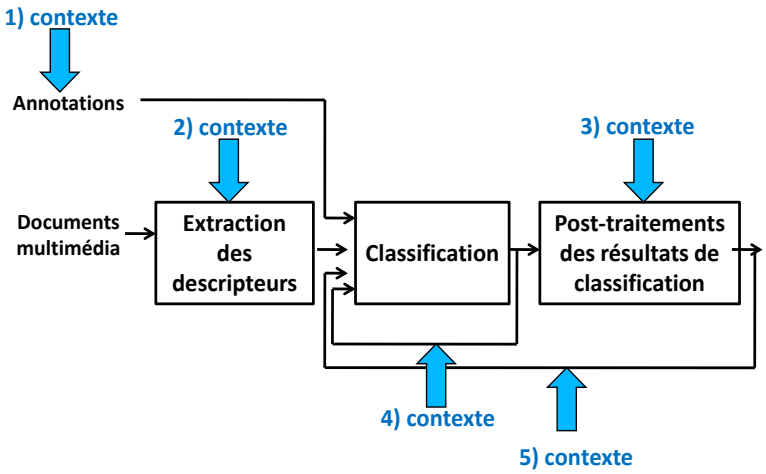


FIGURE 1.6 – Utilisation du contexte dans différentes étapes possibles d'un système d'indexation.

Notre travail de thèse présente plusieurs points forts. Le premier point important est la généricité de nos approches. En effet, **notre travail est fait de manière à atteindre nos objectifs tout en proposant des approches génériques, pouvant être appliquées à n'importe quel système d'indexation pour la détection de n'importe quel concept cible.** On peut voir le système de référence utilisé comme une boite noire, qui peut être remplacée par n'importe quel autre système permettant de détecter des concepts visuels dans des images et/ou vidéos en fournissant un score reflétant la probabilité qu'un

document multimédia contienne le concept cible. D'autre part, nous avons insisté pour que nos contributions ne soient pas spécifiques à une catégorie particulière de concepts, en considérant dans les expérimentations plusieurs types de concepts : personnes, objets, évènements, etc, et en ne pas incluant l'information liée au type de concept dans les approches considérées. Un autre point important de ce travail de thèse est le fait de considérer plusieurs approches agissant à des niveaux différents d'un système d'indexation. Cela permettrait de comparer et trouver le niveau le plus propice pour exploiter efficacement le contexte.

En outre, les méthodes développées sont évaluées dans le cadre des compagnes internationales, notamment TRECVID [3] qui est devenue un standard dans le domaine de l'indexation et la recherche multimédia. Le travail est réalisé dans le contexte du programme Quaero [4]. Tout cela permet entre autres choses d'avoir accès à un grand volume de données images et vidéos annotées. Même si on n'a pas évalué toutes nos contributions avec des soumissions effectives à la compagne TRECVid, nous avons suivi lors de nos évaluations, le même protocole d'évaluation et les mêmes outils imposées par cette compagne d'évaluation.

1.5 Contributions / Plan de la thèse

De nombreuses contributions ont été apportées par ce travail de thèse dans le cadre de l'indexation sémantique des images et vidéos, et spécialement pour l'utilisation des informations contextuelles par les systèmes automatisés d'indexation. Ces contributions valident les objectifs énoncés dans la section 1.4. Nous présentons dans ce qui suit la structure et le contenu de ce mémoire, tout en décrivant nos différentes contributions.

Chapitre 2 - État de l'art : introduit une présentation générale de la notion d'un concept et d'un système classique d'indexation d'images et de vidéos et décrit les différentes méthodes pertinentes pour notre travail ;

Chapitre 3 - Contexte sémantique : présente nos contributions pour l'utilisation du contexte sémantique pour l'indexation des images et vidéos. Trois approches avec des traitements différents dans des niveaux différents d'un système d'indexation sont présentées. La première : "Ré-ordonnancement sémantique basé sur une ontologie", exploite les relations inter-concepts via l'utilisation de la hiérarchie d'une ontologie de concepts et les annotations d'un corpus de données et/ou les scores de détection calculés sur un ensemble d'entrainement. Deux stratégies de sélection des concepts depuis la hiérarchie de l'ontologie sont proposées : "ascendants et descendants" et "Famille de concept" . Cette proposition qui exploite le contexte sémantique est inspirée de [WTS04]. La deuxième : "reclassement sémantique par regroupement", décrit chaque document multimédia (image/vidéo) avec des informations sémantiques. L'approche proposée regroupe les échantillons d'apprentissage en se basant sur leurs contenus sémantiques. Cette méthode

3. http ://www-nlpir.nist.gov/projects/
4. http ://www.quaero.org

s'ajoute à l'ensemble des travaux de l"état de l'art exploitant le contexte sémantique. La troisième : "rétroaction conceptuelle", construit un descripteur conceptuel de haut niveau et l'exploite de la même manière qu'un descripteur de bas niveau. Cette approche appuie le travail de Smith et al. [SNN03], en apportant de nombreuses nouveautés dans la méthode d'exploitation du contexte sémantique via un classifieur. Nous avons apporté plusieurs extensions qui viennent enrichir cette contribution ;

Chapitre 4 - Contexte temporel : présente nos contributions pour l'utilisation du contexte temporel pour l'indexation des vidéos. Deux approches différentes sont présentées et comparées. La première : "re-scoring temporel", est une méthode de ré-ordonnancement des résultats du système de référence qu'on veut améliorer. Cette dernière est une continuité du travail de Safadi et al. [SQ11a]. La deuxième : "Descripteurs temporel", agit dans l'étape d'extraction des descripteurs de bas niveau. L'étude comparative entre les deux approches exploitant le contexte temporel est une nouveauté dans le cadre de notre problématique ;

Chapitre 5 - Combinaison des contextes sémantique et temporel : présente nos contributions concernant l'utilisation conjointe des contextes sémantique et temporel pour l'indexation des vidéos. Deux approches combinant des méthodes présentées dans les deux chapitres précédents sont décrites et évaluées. La première : "re-scoring à deux couches", combine l'approche "re-scoring temporel" avec les méthodes "Ascendants et descendants" et "Famille de concept" dans une approche à deux couches. La deuxième contribution : "rétroaction conceptuo-temporelle", incorpore le contexte temporel dans l'approche "rétroaction conceptuelle". La combinaison des aspects temporel et sémantique n'est pas une idée nouvelle, mais l'originalité de nos propositions s'inscrit dans la proposition de nouvelles approches les combinant dans une étape de post-traitements des résultats de classification ou via un classificateur. Ces contributions présentent de nouvelles façons d'utiliser conjointement les contextes sémantique et temporel ;

Chapitre 6 - Détection simultanée de plusieurs concepts décrit une nouveauté par rapport à ce qui a été présenté dans les chapitres précédents. En effet, dans ce qui précède, le but était de détecter un seul concept dans un document multimédia. Dans ce chapitre, nous abordons la problématique de détection simultanée d'un groupe de concepts au lieu d'un seul. Deux catégories de méthodes sont présentées et plusieurs variantes sont testées et évaluées. Cet axe de recherche est lié à notre problématique. En effet, détecter simultanément un groupe de concepts revient à chercher l'occurrence d'un concept dans un contexte dans lequel les autres sont présents/absents. Nous avons comparé deux grandes familles d'approches et nous les avons évalués sur des corpus d'images et de vidéos ;

Comme nous avons conduit l'évaluation de nos contributions sur les mêmes types de données, nous sommes capables de dire dans quelle mesure et dans quelles conditions chacune de nos approches est qualifiée pour donner de meilleurs

résultats par rapport aux autres. Cela constitue l'une des plus importantes contributions du travail de cette thèse.

Cette thèse a conduit à une publication dans une revue internationale, cinq articles de conférences et ateliers de travail internationaux, et deux publications dans des conférences nationales, comme décrit dans l'annexe C.

Chapitre 2

État de l'art

Ce chapitre présente l'état de l'art du travail de cette thèse. Il commence par un survol de la problématique de définition d'un concept. Ensuite, une description générale d'un système d'indexation de documents multimédia est présentée. Chacune des étapes du système d'indexation est ensuite détaillée : 1) L'extraction de descripteurs, 2) L'optimisation des descripteurs, 3) Les méthodes d'apprentissage/classification, 4) La fusion et 5) L'ordonnancement. La notion du contexte est ensuite introduite, c'est la partie qui est reliée au coeur de cette thèse. Une définition du contexte est exposée, pour ensuite passer aux différents types de contexte et aux méthodes utilisées pour l'incorporer dans un système d'indexation de documents multimédia, tout en faisant un tour à l'horizon sur les travaux de l'état de l'art qui utilisent le contexte. Le chapitre se termine par une section décrivant le processus d'évaluation des systèmes d'indexation avec une présentation de certaines compagnes d'évaluation de l'indexation sémantique des images et vidéos, en finissant par les choix de méthodes et des types de contexte considérés dans le cadre de cette thèse.

2.1 Documents image et vidéo

Une image peut être vue d'un point de vue physique comme un signal 2D continu. La quantité d'informations contenues dans ce signal mesurées en continu est infinie. Le passage à une représentation numérique se fait en réalisant une discrétisation des coordonnées spatiales de ce signal dans les deux dimensions de l'image, et une discrétisation du signal par un échantillonnage (quantification). Une image numérique est donc une grille d'éléments appelés pixels (contraction des mots anglais "picture element", c'est-à-dire élément d'image).

La vidéo, quant à elle, est beaucoup plus complexe qu'une image. D'un point de vue physique (ou informatique), un document (ou un flux) vidéo est une combinaison de sous-médias ou « pistes » organisés suivant un axe temporel. Chaque piste est présente sous la forme d'un flux d'éléments et ces flux sont synchronisés entre eux. Ces différents flux peuvent contenir des images animées, du son (un flux ou une composition de plusieurs flux audio émis en parallèle à une fréquence fixe) ou du texte accompagné des informations permettant de le synchroniser avec les autres flux [CHA05]. Pour analyser une vidéo, on doit donc structurer une grande

quantité d'informations diverses et hétérogènes. Une vidéo peut être décomposée selon différents niveaux de détails [TAP12], comme décrit dans la figure 2.1 :

- Niveau scène : correspond à un groupe de séquences vidéo qui sont homogènes par rapport à un critère sémantique. Une scène doit respecter trois règles de continuité : en espace, temps et action ;
- Niveau plan : correspond à une succession d'images d'une vue continue d'une caméra ;
- Niveau image/image-clé : correspond à l'ensemble d'images représentatif de chaque plan pouvant résumer son contenu ;

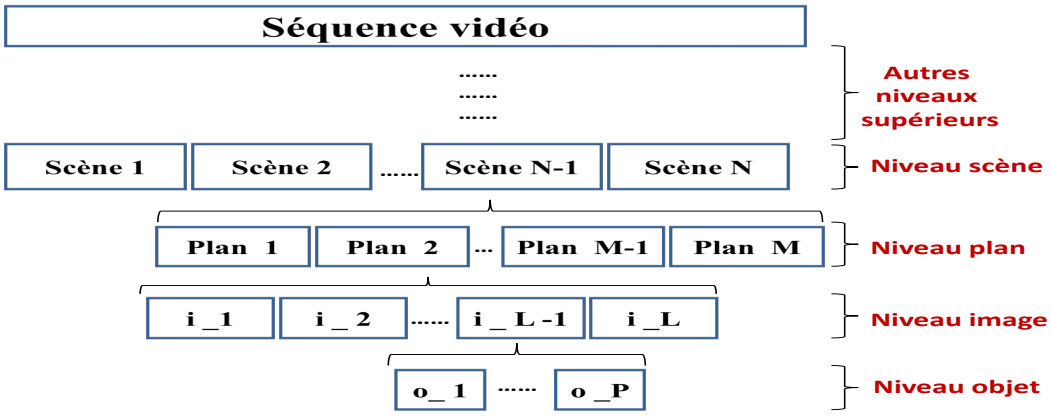


FIGURE 2.1 – Décomposition structurelle d'une vidéo en plusieurs niveaux.

La liste des trois niveaux cités ci-dessus n'est pas exhaustive. On peut peut réaliser des décompositions à un niveau plus élevé en décomposant une séquence vidéo en histoires [HW98], ou à un niveau plus bas : niveau objet, correspondant à des régions spatiales ou spatio-temporelles reliées à des objets saillants. Ce dernier cas peut concerner les images fixes comme dans la tâche d'annotation/indexation de régions d'images [YLZ07].

Dans l'indexation par détection de concept, on peut indexer la vidéo entière, mais cela n'est pas très précis, puisqu'une vidéo peut se voir attribués des concepts qui apparaissent uniquement durant de très courtes périodes. D'autre part, on peut annoter une vidéo au niveau "image", mais cela génère des annotations très denses. De plus, il peut y avoir des images qui contiennent des informations peu importantes, des images redondantes ou des images servant pour référencer un arrêt ou une transition dans la séquence. Par conséquent, les chercheurs choisissent généralement d'indexer les vidéos au niveau "plan". Il existe plusieurs techniques de segmentation de vidéos en plans [BR96]. Ces algorithmes sont généralement basés sur la détection de transition ou d'un changement important dans la séquence.

Pour extraire de bonnes caractéristiques visuelles d'un plan, il n'est pas judicieux d'utiliser toutes les images : elles sont nombreuses et redondantes. Il est donc important de sélectionner la ou les images les plus représentatives du

plan considéré. Cette tâche s'avère difficile. Elle est réalisée d'une manière empirique, en sélectionnant la première, la dernière ou l'image médiane du plan. Plusieurs méthodes sont présentées dans l'état de l'art pour sélectionner les images-clés [Dir00, ZRHM98, ASH96, VPGB02, HM00]. Le nombre d'images-clés à considérer dépend de la nature des vidéos. Par exemple, dans certaines séquences vidéos qui contiennent des mouvements, une seule image-clé n'est pas suffisante pour représenter le plan. Il a été montré que la représentation d'un plan par de multiples images-clés n'implique pas nécessairement une meilleure performance des systèmes d'indexation.

2.2 Notion de concept

Dans l'analyse sémantique des documents, on a tendance à utiliser le terme "concept". En effet, dans le cadre ce travail, comme dans l'indexation des documents, on opte pour une approche de détection de concepts dans les documents. Il est donc important d'illustrer ce que c'est qu'un concept. Le débat concernant la définition d'un concept et sa distinction par rapport au terme ou au mot, persiste depuis longtemps, sans pouvoir arriver à un consensus final.

Basé sur les travaux de Ausubel [Aus68, Aus00] et Toulmin [Tou72], Novak et al. [NG84] définissent un concept comme une régularité ou un motif perçu(e) dans des événements ou des objets, désigné par une étiquette.

Flavel, Miller et Miller (2002) définissent un concept comme un groupement mentale des différentes entités en une seule catégorie sur la base d'une certaine similitude sous-jacente - d'une certaine façon dans laquelle toutes les entités se ressemblent, ou certain tronc commun qui les rend tous, en un certain sens, la même chose.

[ML99] font la distinction entre un concept lexical, un concept primitif et un concept complexe. Selon les auteurs, les concepts lexicaux sont des concepts (e.g. *BACHELOR*, *BIRD*, etc) qui correspondent à des éléments lexicaux dans les langues naturelles. Il est commun de penser que les mots dans les langues naturelles héritent leurs significations des concepts qu'ils expriment. En effet, dans certaines discussions, les concepts sont pris pour être seulement ces représentations mentales qui sont exprimées par des mots dans les langues naturelles. Cependant, cette utilisation est délicate, car elle interdit de considérer comme concepts les représentations qui sont exprimées par des expressions complexes en langage naturel. Deux autres points de terminologie doivent être mentionnés. D'autre part, les auteurs définisse un concept primitif comme celui qui n'a pas de structure. Un concept complexe est, en revanche, celui qui n'est pas primitif.

Il existe plusieurs théories tournant autour de la notion de concept. La plus célèbre est " la théorie classique". La théorie classique est appelée ainsi puisqu'elle est celle défendue par Platon, Decartes ou encore Kant [SM81]. Elle définie un concept selon les caractères qu'il possède. Ces caractéristiques sont censées être nécessaires et suffisantes pour déterminer l'appartenance ou non d"un élément à une catégorie conceptuelle. Cependant, la théorie classique ne dit pas sur quelle base ces caractères sont déterminés. Pour Platon ou Kant, les caractères sont

ceux qui font que l'élément est ce qu'il est. Par exemple, une autruche a des ailes, ne vole pas, a de grandes pattes, court vite, etc. Kant a introduit la notion de catégorie pour regrouper des concepts. La structure de la classification peut alors être arborescente, ce qui rend alors compte de la hiérarchie des concepts. Ainsi, on peut affirmer que le concept "baleine" est inclut dans le concept "mammifère", et donc que toutes les baleines sont nécessairement des mammifères, mais la réciproque n'est pas nécessairement vraie. L'avantage est que cette théorie explique très bien ce qu'est un concept. Un objet tombe sous un concept et un seul du fait des propriétés qui le caractérisent. Ainsi, comme l'a montré Kant, le concept est souvent considéré comme forme de catégorisation. Dire que les éléments a_1, a_2, $\ldots$,a_n relèvent du concept C revient d'un point de vue ensembliste à dire que les éléments a_1, a_2, $\ldots$,a_n sont inclus dans l'ensemble étiqueté par C [Aya07]. D'autre part, il existe pour Kant des concepts empiriques [Kan01] ou a posteriori qui sont tirés de l'expérience des sujets. Ces concepts s'opposent aux concepts a priori, c'est-à-dire indépendants de toute donnée sensible [Aya07].

D'une manière ou d'une autre, la plupart des théories de concepts peuvent être considérées comme des réactions à, ou de l'évolution de, ce qui est connu comme la théorie classique des concepts [ML99]. La théorie classique soutient que la plupart des concepts, surtout les concepts lexicaux ont une structure définitionnelle. Cela signifie que la plupart des concepts codent des conditions nécessaires et suffisantes pour leur application. Considérons, par exemple, le concept "Célibataire" (Bachelor en anglais). Selon la théorie classique, on peut voir ce concept comme une représentation mentale complexe qui spécifie les conditions nécessaires et suffisantes pour que quelque chose soit "Célibataire". Donc, le concept "Célibataire" pourrait être composé d'un ensemble de représentations telles que : "n'est pas marié", "est un homme" et "est un adulte", etc. Chacun de ces éléments indique une condition qu'une chose doit satisfaire afin d'être "Célibataire".

Dans le cadre de notre thèse, nous évitons ce genre de débats philosophiques et nous optons pour une approche pragmatique. Nous considérons un concept comme un terme sémantique défini via une description accompagnée d'un ensemble d'exemples visuels. On peut aussi considérer des concepts de bases du genre : personne, bus, car, chien, etc, et définir d'autres concepts à base de l'ensemble de concepts de base considéré. C'est finalement l'expert humain qui décide des définitions des concepts. Par exemple, une "fenêtre" peut être définie comme : *une ouverture dans la paroi ou le toit d'un bâtiment ou d'un véhicule équipé de verre ou autre matériau transparent.* Une "acclamation" est définie comme :*une ou plusieurs personnes en train d'acclamer ou applaudir.* Ces deux définitions sont considérées dans la compagne internationale TRECVid (voir la section 2.13.2.3).

2.3 Indexation par le contenu des documents multimédia

Le développement des dispositifs d'acquisitions d'images, des capacités de stockage, la baisse des coûts des matériels informatiques et la disponibilité des techniques de numérisation de haute qualité que nous observons ces dernières années, se traduit par une production permanente et considérable d'images numériques dans différents domaines, ce qui a conduit à un développement constant des bases de données d'images et vidéos, surtout sur le web où des milliers de données sont mises en ligne par heure. La nécessité d'un système de traitement et d'analyse automatique de ces bases de données multimédia n'est donc plus à démontrer. Plusieurs systèmes d'indexation et de recherche de documents multimédia par le contenu ont vu le jour, la majorité d'entre eux concernent les images. Un système typique de recherche d'images par le contenu permet aux utilisateurs de formuler des requêtes en présentant un exemple du type de l'image recherchée, bien que certains offrent d'autres solutions telles que la sélection dans une une palette ou une liste d'exemples. Le système identifie alors parmi la collection d'images celles qui correspondent le plus à l'image requête, et les affiche. La correspondance entre l'image requête et l'ensemble des images de la base de données se fait en comparant les caractéristiques de bas niveau des mages. Ces caractéristiques sont des mesures mathématiques de la couleur, texture et/ou de forme, etc. En ce qui concerne les autres modalités, plusieurs systèmes se servent de l'audio comme source d'information quand cette modalité s'avère une source potentielle d'information utile. À titre d'exemple, dans les applications dotées d'un système de reconnaissance vocale, la voix d'une personnes est extraite via un dispositif d'enregistrement, puis comparée avec des éléments d'une base de données en comparant les caractéristiques de bas niveau, dans le but de reconnaitre le locuteur, pour lui permettre d'effectuer ou non un ensemble d'opérations liées à l'application. D'autre part, il existe aussi des systèmes opérant sur des vidéos, comme par exemple, les applications de détection de copies. Étant donnée, une séquence vidéo, ce genre de systèmes vise à présenter l'ensemble des vidéos qui contiennent la même séquence qu'une vidéo en entrée, ou juste une partie, ou extraire les séquences en commun entre deux vidéos. Le même principe existe dans les systèmes de détection de plagia dans les documents textuels. Le tableau 2.1 présente quelques exemples de systèmes d'indexation et de recherche d'images et de vidéos qui ont vu le jour ces dernières années.

Malgré leur utilité, ce genre de systèmes ne répondent pas à tous les besoins des utilisateurs. Nous rappelons que ces systèmes ignorent totalement la sémantique et fonctionnent de manière aveugle, en comparant des caractéristiques de bas niveau. La manière la plus simple pour un utilisateur, surtout un novice ou débutant, est de formuler ses attentes à travers des termes textuels. Pour ce faire, un système automatique doit pouvoir faire une correspondance entre du texte compréhensible par l'humain et un contenu multimédia. Autrement dit, il devient nécessaire de passer à une analyse sémantique. Cela constitue un des majeurs défis de la recherche d'information multimédia par le contenu. On distingue

Systèmes	Propriétés
QBIC [FEF+94]	Développé par IBM, Premier système commercial de recherche d'image par le contenu, La requête est formulée grâce à la recherche par l'exemple,Utilise la couleur, la texture, etc. http ://www.qbic.almaden.ibm.com
Virage [BFG+96]	Développée par Virage Inc, Similaire à QBIC, Permet la combinaison de plusieurs types de requêtes, L'utilisateur peut attribuer des poids pour chaque mode, http ://www.virage.com/
Netra [DMM98]	Développé dans le projet UCSB Alexandria Digital Library, Utilise la couleur sur des régions pour chercher des régions similaires dans la base de données, La version 5 utilise le mouvement dans la segmentation spatio-temporelle, http ://vivaldi.ece.ucsb.edu/Netra
Informedia [WKSS96]	Développé par l'université Carnegie Mellon, Exploite le mouvement de la caméra et l'audio, Réalise une reconnaissance vocale automatique, http ://www.informedia.cs.cmu.edu/
Photobook [PPS96]	Développé par MIT Media Laboratory, Se base sur 3 critères (couleur, texture et forme), Utilise plusieurs méthodes (distance euclidienne, mahalanobis, divergence, histogrammes, vecteurs d'angle, etc). La version améliorée permet la combinaison de ces méthodes, http ://www-white.media.mit.edu/vismod/demos/photobook/

TABLE 2.1 – Quelques exemples de systèmes d'indexation et de recherche d'images et de vidéos. Descriptions tirées de la thèse de R. Benmokhtar ([Ben09]).

deux types d'approches de recherche d'information multimédia par le contenu. Cette distinction est faite en fonction de deux principaux niveaux de contenu sur lesquels ils opèrent :

- Bas niveau : ce genre d'approches se base sur la comparaison de caractéristiques de bas niveau (couleur, texture, etc). Elles ignorent par contre la sémantique, et se contentent d'utiliser des mesures de similarité/dissimilarité entre deux documents multimédia.
- Haut niveau : niveau sémantique, interprétable et compréhensible par l'humain. En plus des approches opérant uniquement sur un contenu bas niveau, ces approches manipulent des sémantiques relatifs au documents. En décrivant les documents avec des caractéristiques de bas niveau et des sémantiques (termes, concepts, ...), ils deviennent accessibles de deux manières différentes : 1) via une comparaison des caractéristiques de bas niveau, et/ou 2) par un filtrage par mots clés. Bien que ces approches semblent plus intéressantes, elles présentent plusieurs difficultés présentées dans la suite.

Il est utile de décrire chaque document multimédia par des termes/mots/concepts pour rendre ces documents exploitables pour un usage donné, spécialement par un système de recherche par mots clés. On appelle “indexation”, le fait d’assigner un ou ensemble d’indexes à un document. Un index peut être de différents types : des symboles, des concepts, des termes, des sémantiques, etc. Dans le cadre de la recherche d’information multimédia, on assigne aux documents des concepts, ou termes sémantiques. Ce processus appelé “indexation sémantique” peut être réalisé de trois manières différentes :

1. Manuelle : l’opération est effectuée par l’intervention d’un expert humain pour attribuer à chaque document multimédia un ou un ensemble de concepts/sémantiques qui lui sont associés. Cette tâche est souvent considérée comme étant un travail laborieux qui nécessite l’intervention de l’opérateur humain et qui dépend d’un processus totalement manuel. Avec l’explosion de la masse de données multimédia, cette méthode devient de plus en plus impossible à réaliser de façon entièrement manuelle ;
2. Automatique : le processus est réalisé par une machine. Bien que cette méthode soit applicable sur une grande masse de données, la qualité de indexation est très insuffisante pour avoir des recherches précises et efficaces, en l’état actuel des connaissances [1]. Cette mauvaise qualité s’explique par les grands problèmes auxquels se heurte l’indexation automatiquement (voir la section 1.2) ;
3. Semi-automatique : cette solution intermédiaire entre les deux précédentes, dans laquelle un humain intervient une ou plusieurs fois durant le processus d’indexation, soit pour annoter ou raffiner les résultats renvoyés automatiquement ;

On utilise parfois le terme “annotation” au lieu de “indexation”. Mais généralement le terme “annotation” est utilisé pour faire référence à l’indexation manuelle. De ce fait, on parle d’annotation manuelle et d’indexation automatique. On pourra parler, surtout dans ce manuscrit, de détection d’un concept dans un document, qui signifie l’indexation automatique d’un document par le concept en question. Bien que l’efficacité et la précision sont souvent reliées à l’opérateur humain, la rapidité (temps d’exécution) est une caractéristique de la machine. En outre, avec l’exposition de la masse de données, sans parler du coût très élevé (en termes d’argent et surtout de temps) d’une annotation manuelle, il devient de plus en plus impossible d’annoter les documents manuellement. C’est pour toutes ses raisons que la recherche dans ce contexte se focalise sur l’automatisation de la tâche d’indexation. L’indexation automatique se heurte à plusieurs problèmes qui sont très difficiles à résoudre, certains dépendent du problème de l’indexation automatique en lui même, comme par exemple, le problème du “fossé sémantique” présenté dans la section 1.2.1, quant à certains autres sont liés aux approches utilisées pour l’indexation automatique.

Un aspect caractéristique de l’indexation de la vidéo par rapport aux autres types de documents est la présence d’informations provenant de différents canaux : visuel, audio et textuel. Ces trois modalités sont généralement traitées

1. Date de rédaction de la thèse : Juin 2014

séparément, et ce, parce qu'elles concernent des communautés scientifiques différentes. De plus, les outils d'analyse multimédia sont généralement destinés à une seule modalité. En effet, les applications de recherche dans le web utilisent uniquement du texte. Les applications de recherche d'images combinent à peine le texte et le contenu visuel. Les logiciels d'analyse de vidéos agissent séparément sur le son et l'image. Cependant, les approches actuelles tendent à considérer les différentes modalités dans une même approche multimodale [BMM99, CSS98]. Cela est réalisé en faisant une combinaison des résultats des traitements effectués sur les différentes modalités.

2.3.1 Description d'un système d'indexation multimédia

L'indexation sémantique de documents multimédia est généralement réalisée par détection de concepts visuels via des approches d'apprentissage automatique supervisée. Un classificateur est entrainé sur un ensemble de données annotées manuellement par rapport un ensemble fini C de classes cibles $c_k \in C$. Pour chaque échantillon de l'ensemble d'entrainement, un ou plusieurs descripteurs de bas niveau sont extraits et décrits dans un espace F. Chaque exemple i est représenté par un couple (x_i, y_i), où $x_i \in F$ est son descripteur de bas niveau, et $y_i \in \{0, 1\}$ est son annotation manuelle par rapport à la classe cible c_k, où 1 signifie : le concept c_k est présent dans l'échantillon x_i et 0 signifie : l'échantillon x_i ne contient pas le concept c_k. On parle ici d'une classification binaire (deux valeurs). L'ensemble d'entrainement X est défini donc comme suit : $X = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, où n est le nombre d'échantillons contenus dans X et chaque couple $(x_j, y_j) \in F \times \{0, 1\}$. Une règle de classification est apprise de façon à assigner à chaque nouvel échantillon x_j la classe c_k si la probabilité $P(c_k|x_j)$ dépasse un seuil donné. Nous rappelons qu'à ce stade, nous parlons d'une classification binaire. Dans le cas où une classification multi-labels est considérée, l'échantillon x_j se verra attribué la classe $c \in C$ qui maximise la probabilité $P(c|x_j)$.

Mathématiquement parlant, le but est d'apprendre une fonction via un algorithme d'apprentissage automatique L défini comme suit :

$$\begin{array}{rcccl} L & : & (X \times C)^* & \rightarrow & C^{||X||} \\ & & S & \mapsto & g = L(S) \end{array} \tag{2.1}$$

où $S = \{(x_i, y_i), 1 \leq i \leq n\}$: est l'ensemble d'entrainement de taille n.

Après avoir consulté un certain nombre d'échantillons S, l'apprenant doit généraliser son modèle sur des exemples non encore vus. Le but final est d'apprendre une fonction cible f pour chaque classe cible $c_k \in C$:

$$\begin{array}{rcccl} L & : & X & \rightarrow & \mathbb{R} \\ & & x & \mapsto & f(x) = P(c_k|x) \end{array} \tag{2.2}$$

La valeur $f(x)$ reflète la probabilité que x appartient à la classe c_k. Or, cette valeur n'est pas nécessairement une probabilité, mais un score jouant le rôle de

probabilité, servant à classer l'exemple x dans la classe c_k qui a le meilleur score (plus petit ou plus grand, dépendant de la méthode du calcule et du sens de la valeur calculée).

Comme nous l'avons déjà mentionné, généralement, la classification est réalisée de façon binaire. L'annotation d'un document par rapport à une classe peut alors prendre deux valeurs différentes : 1) L'exemple est positif, c'est-à-dire que le concept est présent dans document, 2) L'exemple est négatif : le concept est absent du document. Le processus est déroulé très souvent pour chaque concept indépendamment des autres. Certaines approches tentent quant à elles, d'apprendre un modèle multi-classes et considèrent dans ce cas les annotations des exemples d'entrainement par rapport à un ensemble de classes et pas uniquement une seule [QHR+08, BLSB04].

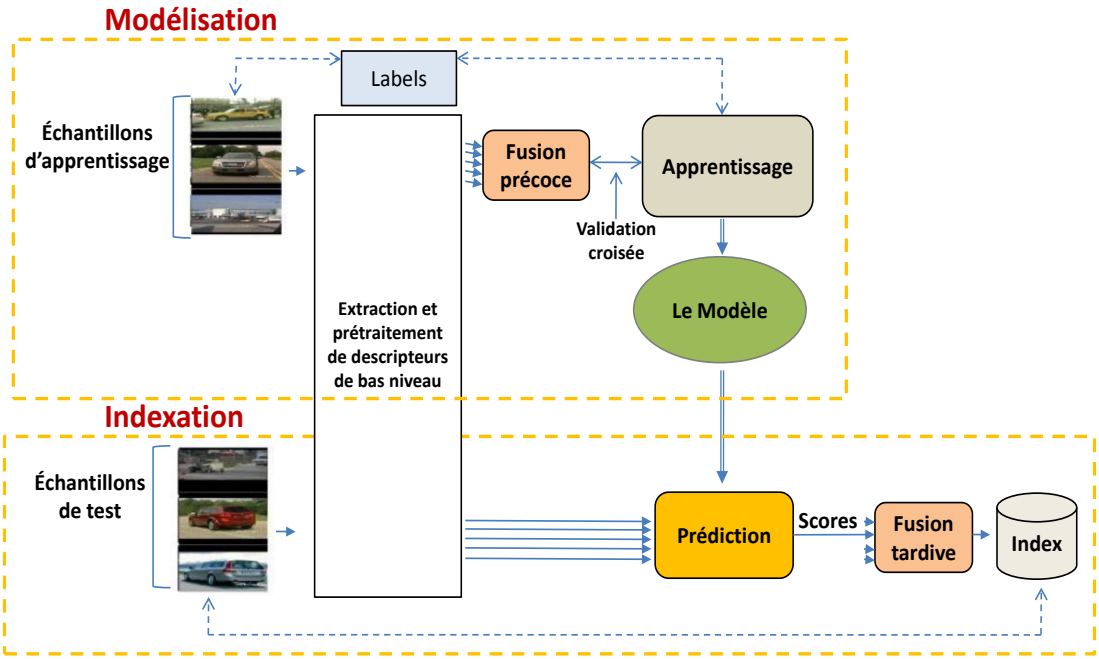


FIGURE 2.2 – Architecture d'un système d'indexation de documents multimédia.

La figure 2.2 présente un schéma général décrivant un système d'indexation de document multimédia pour un concept cible donné (i.e., les annotations concernent ce concept cible). On peut constater que le processus d'indexation passe par deux étapes : 1) La modélisation ou l'apprentissage, 2) L'indexation ou la prédiction. Dans l'étape de modélisation, le système cherche à étudier la corrélation entre les descriptions des exemples et leurs annotations dans le but de créer un modèle de classification. Le modèle est ensuite utilisé pour assigner des scores de classification à des exemples étant donnés leurs descripteurs de bas niveau. Ces scores sont interprétés comme la probabilité que l'exemple en question contienne le concept cible. Avant que le modèle soit appris, un ou plusieurs descripteurs sont extraits de chaque échantillon d'apprentissage (voir la section 2.4). Une étape de pré-traitement et d'optimisation de descripteurs peut être prévue (voir la section 2.6). L'étape d'optimisation peut concerner la normalisation ou la réduction de dimensionalité. Cette étape peut influencer l'étape

d'apprentissage, et donc la performance du système. Une fois les descripteurs prêts, on peut réaliser une fusion de ces descripteurs. Deux types de fusions sont possibles : la fusion précoce et la fusion tardive (voir la section 2.8). L'apprentissage est donc lancé à ce stade en utilisant un algorithme d'apprentissage supervisé (e.g. SVM, Réseaux de neurones, arbres de décisions, ...). L'algorithme d'apprentissage utilise les descripteurs ainsi que les annotations manuelles des échantillons d'apprentissage, pour apprendre une fonction de classification. Dans l'étape d'indexation ou de prédiction, le modèle appris dans la phase d'apprentissage est appliqué sur des échantillons non vus (l'ensemble de test), pour calculer des scores de prédiction reflétant la probabilité que ces exemples contiennent le concept étudié (à détecter). Les échantillons de test sont ensuite ordonnés par ordre décroissant par rapport à leurs scores de prédiction. L'élément ayant plus de chance ou de probabilité de contenir le concept cible se retrouvera en haut de la liste. La liste ordonnée est ensuite retournée comme résultat du processus d'indexation ou de recherche. Nous notons ici que l'étape d'ordonnancement ne sert que pour la visualisation ou le calcul de la mesure d'évaluation (précision moyenne) qui nécessitent un ordonnancement des documents en fonction de leur susceptibilité de contenir le concept cible. Les mêmes descripteurs sont extraits des échantillons de test et ces derniers subissent les mêmes approches d'optimisations avant de passer à l'étape de prédiction.

Chacune des deux parties d'apprentissage et de prédiction nécessitent plusieurs traitements ou des sous-étapes, notamment l'étape d'extraction de descripteurs de bas niveau et la normalisation de ces descripteurs. Le système peut aussi contenir des étapes de fusion de descripteurs ou de résultats de différentes classifications, comme le montre la figure 2.2. Nous allons détailler par la suite ces différentes étapes. Nous aborderons l'évaluation de systèmes d'indexation dans la section 2.13.

2.4 Description de documents image et vidéo

Il n'existe pas une définition précise d'un descripteur, tout dépend du problème et du type d'application. Un descripteur concerne en général une partie ou des points d'intérêts d'une image. Les descripteurs sont en général utilisés en entrée des algorithmes de la vision par ordinateur, cela rend l'efficacité de ces algorithmes relative à la qualité des techniques d'extraction des descripteurs utilisés. Un bon descripteur est celui qui décrit le contenu avec une grande variance pour être capable de distinguer tout type de média, en prenant en compte la complexité de l'extraction, la taille du descripteur et l'échelle de l'interopérabilité (capacité que possède un système intégralement connu, à fonctionner avec d'autres systèmes existants ou futurs). En outre, un bon descripteur doit permettre de reconnaitre le contenu même en cas de certaines variations : changement d'illumination, variation de l'échelle, translation et rotation, changement de points de vues, ... Pour absorber l'effet de ces transformations, il existe des descripteurs invariants [Rot95] ou quasi-invariants [BL93] à ces transformations, c'est-à-dire que deux images qui se différencient par une de ces transformations auront des descripteurs simi-

laires (voire identiques). Les descripteurs peuvent être invariants aux changement d'illumination [FDF94, SH96], à certaines transformations géométriques liées aux changements de point de vue [MA92, MZF93], ou aux deux à la fois [vGMU96].

2.4.1 Descripteurs locaux et descripteurs globaux

On peut utiliser des descripteurs caractérisant la totalité de l'image (descripteur global) ou plusieurs descripteurs locaux caractérisant chacun une partie de l'image. Les techniques modernes en imagerie tendent à privilégier les descripteurs locaux aux globaux car les descripteurs locaux sont plus efficaces et ils permettent une recherche plus fine et absorbent mieux certaines variations. Dans la cas de descripteurs globaux, un seule descripteur décrit la totalité de l'image, cela les rend robustes au bruit qui peut affecter le signal, les histogrammes de couleur et des niveaux de gris en sont des exemples classiques [SS94], mais d'autres descripteurs existent comme le corrélogramme [HKm+97] et les angles de couleurs [FCF96]. L'inconvénient de ces descripteurs est qu'ils ne permettent pas de distinguer des parties de l'images, ils ne distinguent pas, par exemple, les objets dans l'image, sauf dans le cas où l'image ne contient qu'un seule objet dans un fond uni. Par opposition, les descripteurs locaux s'associent à une partie/région de l'image qu'on commence par détecter avant de calculer le descripteur, cette partie peut concerner un objet par exemple, la détection se fait indépendamment de la position dans l'image, ce qui assure l'invariance par translation.

On peut également catégoriser les descripteurs selon le type de modalité qu'ils représentent : descripteurs visuels, descripteurs de l'audio, descripteurs de mouvement, etc. Nous détaillons ces points dans les sections suivantes.

2.4.2 Descripteurs visuels

2.4.2.1 Descripteurs de couleur

Les descripteurs de couleurs sont les plus utilisés dans le domaine de la recherche des images et des vidéos par le contenu. [vdSGS08b] présente une comparaison entre plusieurs descripteurs de couleurs dans le cadre de la détection de concepts dans les images et vidéos. L'histogramme s'avère le descripteur le plus simple à calculer, son calcul consiste à compter le nombre d'occurrences des différentes valeurs possibles d'intensité des pixels dans l'image. On peut distinguer plusieurs catégories d'histogrammes, on peut les classer, par exemple selon l'espace de couleur considéré lors du calcul : "histogramme RGB", "histogramme HSV", "histogramme Opponent", associés respectivement aux espaces de couleurs : "RGB", "HSV", "Opponent color space". "L'histogramme *rg*" est décrit dans le modèle de couleur RGB normalisé (r+g+b=1), les composantes r et g représente l'information sur les couleurs dans une image (b est redondant du fait que r+g+b=1) : r= r/(r+g+b), g= g/(r+g+b) , b= b/(r+g+b). À cause de la normalisation, les composantes r et g sont invariantes à l'échelle et ainsi invariantes par rapport aux changements de l'intensité de lumière et ombres. [PZ99] propose un histogramme conjoint de deux images.

[MC09] définissent “le descripteur de couleurs dominantes”, qui spécifie un ensemble de couleurs dominantes dans l'image. Ce descripteur est obtenu en regroupant les couleurs d'une image en un nombre réduit de couleurs dominantes :

$$F = \{(c_i, p_i, v_i), s\},\ i = 1, ..., N \tag{2.3}$$

p_i : pourcentage de pixels dans l'image
v_i : variance associée à la couleur c_i
s : la cohérence spatiale :nombre moyen de pixel en connexion avec une couleur dominante.

Stricker et Orengo [SO95] représentent la couleur de manière très compacte par un vecteur contenant la moyenne, la variance et le coefficient d'asymétrie (i.e. : respectivement, les moments d'ordre 1, 2 et 3). Ce descripteur a des limites, à partir du moment ou il ne code aucune information spatiale, il devient donc statistiquement possible d'avoir des moments proches pour des images très différentes et qu'il est de plus impossible d'avoir une description des couleurs présentes dans l'image à partir de ce descripteur.

Le corrélogramme [HKm+97] recherche des motifs dans un voisinage donné. Il est assimilable a une matrice $(n \times n \times r)$ où n est le nombre de couleurs utilisées et r la distance maximale du voisinage considéré. Dans cette matrice, la valeur en (i, j, k) désigne la probabilité de trouver un pixel de couleur i à une distance k d'un pixel de couleur j. La représentation se fait le plus souvent par un vecteur résultant de la concaténation des lignes de la matrice.

Décrit par Pass et al. [PZM96], le vecteur de cohérence se propose de séparer les couleurs “cohérentes” des couleurs “incohérentes”. Par ”cohérentes”, sont désignées les couleurs qui sont dans une zone spatiale de couleurs voisines.

2.4.2.2 Descripteurs de texture

La texture représente également un descripteur bas niveau efficace utilisé dans le cadre de l'indexation et la recherche par le contenu. Plusieurs techniques ont été développées pour mesurer la similarité de textures. La majorité des techniques comparent les valeurs de ce qui est connu par les statistiques du second ordre, calculées à partir des images requêtes. Ces méthodes calculent les mesures de textures d'images comme étant le degré de contraste, la grossièreté, la directivité et la régularité [TMY76, ea93] ; ou de la périodicité, la directivité et l'aspect aléatoire [LP96]. D'autres méthodes d'analyse de texture pour la recherche d'images incluent l'utilisation de filtres de Gabo [MM96b] et les fractales [Kap98].

Dans [TJ98], l'auteur présente 4 « familles » d'outils de caractérisation de la texture. On distingue parmi elles les méthodes statistiques, les méthodes géométriques, les méthodes à base de modèles probabilistes et les méthodes fréquentielles.

En utilisant des filtres prédéfinis, la méthode de [LAW80] utilise des convolutions spatiales pour construire 25 versions d'une image texturée, chaque version décrit une caractéristique précise de l'image.

Le filtre de Gabor : Le filtre de Gabor (ou ondelettes de Gabor) est largement adopté pour extraire les caractéristiques de textures à partir des images

pour la recherche d'images [MM96a, Smi97, Den99, Ma97, (ed00, Dim99], et on a montré dans plusieurs travaux que ce descripteur est très efficace. L'utilisation d'un banc de filtres de Gabor permet d'extraire de l'image considérée des informations pertinentes, à la fois en espace et en fréquence, relatives à la texture [BMG90]. En effet, plusieurs recherches conduites (dans [JF91] par exemple) montrent que les fonctions de Gabor simulent de manière convenable le système visuel humain en reconnaissance des textures ; le système visuel étant considéré comme un ensemble de canaux de filtrage dans le domaine fréquentiel. La convolution de l'image par les filtres de Gabor peut se faire dans le domaine spatial ou fréquentiel.

[MOVY01] définissent "un descripteur de texture homogène" qui fournit une caractérisation quantitative de la texture de l'image. Ce descripteur est calculé par filtrage de l'image avec une banque des filtres sensibles à l'orientation et à l'échelle, et en calculant ensuite la moyenne et l'écart-type des sorties filtrées dans le domaine fréquentiel. Précédemment des travaux approfondis [Fcd01, HM99, LSCB99, MM96b, Ro98] sur ce descripteur ont montré que ce descripteur est robuste, efficace et facile à calculer.

[MWNS00] proposent un descripteur compact qui ne nécessite que 12 bits (maximum) pour caractériser la régularité d'une texture (2 bits), la directivité (3 bits $\times$ 2), la grossièreté(2 bits $\times$ 2). Ce descripteur est utile pour les applications de navigation (Browsing), et est utilisé conjointement avec le descripteur de texture homogène, il peut aider à une recherche rapide et précise des images.

2.4.2.3 Descripteurs de formes

Les descripteurs de formes permettent, comme leur nom l'indique, de présenter une information pertinente sur le contenu de l'image et précisément sur la forme. Il existe différents types de descripteurs de formes qui se différencient par leur simplicité/complexité. Il existe plusieurs descripteurs de formes comme : CSS (Curvative Scale Space descriptors) [FMK96, ZL03], les filtres de convolution [TFMB04, Rus02], les descripteurs de fourrier [RC96, ZL01], les moments de Hu et de Zernike [Hu62, CLE05]. Nous n'allons pas détaillé ce type de descripteurs parce que nous ne les avons pas utilisés dans notre travail.

2.4.2.4 Descripteurs basés sur les points d'intérêts

L'extraction des descripteurs visuels sur l'image entière (descripteurs globaux) permet de réduire le nombre de calculs nécessaires, la taille de la base de données ainsi que le coût de recherche des images les plus similaires. Cependant, l'approche globale ne permet pas une recherche efficace d'objets (au sens large) dans l'image. Á l'inverse, les descripteurs extraits d'une partie de l'image (descripteurs locaux) sont efficaces, mais coûteux. Les descripteurs locaux peuvent être des régions de l'image obtenues soit par segmentation de l'image entière (par recherche de régions d'intérêt) ou par recherche des points d'intérêt. Les points d'intérêt d'une image sont les points qui seront trouvés similaires dans les images similaires. Une manière de les déterminer est de prendre en compte les zones où le signal change.

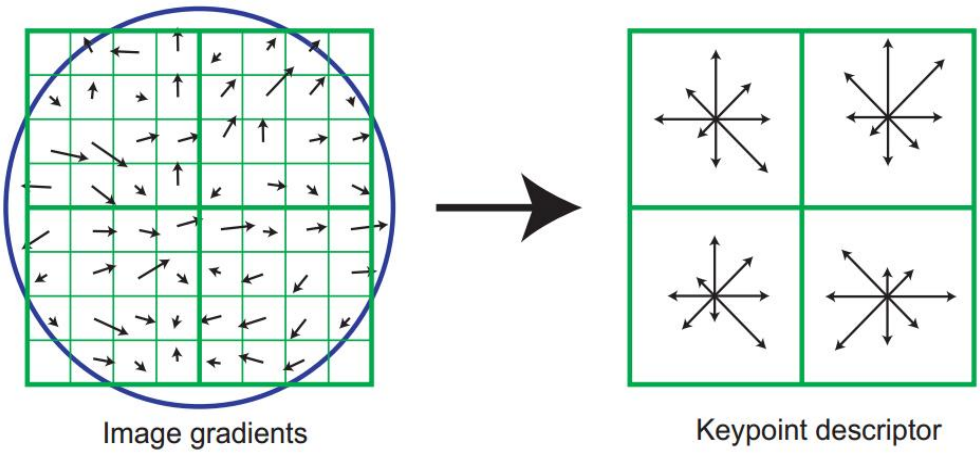


FIGURE 2.3 – principe de calcul des descripteurs SIFT. Image de prise de [Low04].

Par exemple, les points d'intérêt peuvent être les coins, les jonctions en T ou les points de fortes variations de texture.

SIFT : Lowe [Low04] propose des descripteurs appelés SIFT (Scale Invariant Feature Transform), qui sont particulièrement utiles grâce à leur grande distinction dans le cadre de la reconnaissance. Ils sont obtenus en construisant un vecteur de grande dimension représentant les gradients dans une région local de l'image. Les points d'intérêt de l'image sont calculés en utilisant un détecteur (le détecteur de Harris par exemple) à partir desquels les descripteurs SIFT seront calculés. En considérant un point d'intérêt P, le voisinage de P est décomposé en 16 blocs de 4x4 pixels (figure 2.3). Dans chaque bloc un histogramme d'orientation de gradients est formé, en discrétisant l'orientation en 8 bins (correspondant aux différentes orientations possibles :$\{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\}$). Le vecteur représentatif du descripteur aura dans ce cas 128 composantes (4x4x8=128). Les descripteurs SIFT se montrent invariant à l'échelle et la rotation et robuste au bruit et au changement de l'illumination. Les travaux réalisées dans le cadre de l'indexation de documents multimédia ont montré que ces descripteurs donnent pratiquement les meilleurs résultats, malgré leur grande dimension.

2.4.3 Descripteurs audio

L'audio constitue une source d'information utile et importante dans le contexte de l'analyse sémantique des vidéos, surtout pour la détection de concepts qui sont plus faciles à détecter par leurs caractéristiques audio que l'aspect visuel, en particuliers des concepts relatifs à des évènements. En effet, le fait de détecter un dialogue entre personnes, ou des personnes en train de pleurer ou rire, ou une explosion, peuvent aider à détecter les concepts : interview, dispute, fête, attentat, etc.

Dans une approche de recherche sémantique de vidéo en utilisant l'analyse de l'audio, l'audio peut être automatiquement catégorisé en classes sémantiques,

comme “explosion”, “musique”, “parole”, etc [BL02]. Généralement la caractéristiques audio sont regroupées en deux catégories principales [WHL+99] : les coefficients temporels, qui sont calculées directement à partir du signal audio, et des coefficients de fréquence. Les coefficients spectraux sont directement dérivés d'une transformée de Fourier ; ils décrivent le spectre d'un signal audio et sont souvent utilisés pour l'analyse sémantique de contenu audio [LJZ01, ZK98]. Ces coefficients sont couramment utilisés pour la caractérisation et la reconnaissance des haut-parleurs. Parmi eux, les MFCCs (Mel Frequency Cepstrum Coefficients) présentent l'avantage de tenir compte des propriétés non linéaires de la perception humaine de sons selon plusieurs fréquences. Les MFCCs sont couramment utilisés comme descripteur dans les systèmes de reconnaissance de la parole, tels que les systèmes qui peuvent automatiquement reconnaître les numéros dits dans un téléphone. Ils sont également fréquents dans la reconnaissance du locuteur, dans lequel le but est de reconnaître les personnes à travers leurs voix [GFK05].

Le volume sonore, ou plutôt la variation temporelle du volume, reste une des caractéristiques temporelles la plus utilisée, qui peut être un bon indicateur de silence par exemple. En effet le silence est une information sur laquelle se basent certaines approches de segmentation de l'audio [WHL+99]. Le taux de passage à zéro est aussi utilisé comme information dans plusieurs types d'analyses de l'audio, comme par exemple : la reconnaissance de la parole, la recherche d'information musicale et la distinction des dialogues dans la musique, etc. D'autre part le volume peut être une source d'information permettant de détecter la voix qui se caractérise par un faible volume et un grand taux de passage à zéro.

Pour l'analyse du son et plus spécifiquement la musique, on a tendance à utiliser aussi ce qu'on appelle le “pitch”, qui peut être défini comme la fréquence fondamentale du signal audio. Le pitch est considéré comme un attribut majeur auditif de tons musicaux, et est un paramètre important pour l'analyse et la synthèse du son et de la musique. Il est également souvent utilisé pour caractériser d'autres types de sons [WHL+99].

2.4.4 Descripteurs de mouvements

Ces dernières années plusieurs approches de capture ou d'indexation du mouvement ont vu le jour. Certaines consistent à calculer les paramètres de mouvement d'un modèle prédéfini par une technique d'estimation globale, comme le modèle de translation, qui a été adopté par le standard MPEG-7 [ISO01]. D'autres sont basées sur la trajectoire [CCM+98]. Certaines autres utilisent la sémantique, elles fournissent des évènements sémantiques ou des actions, comme le mouvement de la caméra dans la détection des évènements [HQS00].

Un système d'indexation de vidéos basé sur la trajectoire du mouvement de l'objet a été proposé dans [SZ99] où les projections normalisées sur les deux axes de la trajectoire sont traitées séparément avec une transformée en ondelettes en utilisant des ondelettes de Haar. Chen et al. [CC00] segmentent chaque trajectoire en sous-trajectoires en utilisant des coefficients d'ondelette à petite échelle. Un vecteur est extrait à partir de chaque sous-trajectoire et la distance euclidienne entre chaque sous-trajectoire de la trajectoire requête et toutes les

sous-trajectoires indexées sont calculées pour générer une liste de trajectoires similaires dans la base de données. Bashir et al [BKS03, BKS07] ont proposé une approche basée sur l'analyse en composantes principales (voir la section 2.6.2.1) pour l'indexation et la recherche de la trajectoire de mouvement d'un objet qui s'avère très efficaces dans le cas d'une seule trajectoire. Dans le cas où plusieurs objets en mouvement sont considérés, des trajectoires multiples des déplacements d'objets sont extraites en utilisant certains algorithmes de suivi ou des dispositifs de détection du mouvement.

Dans [ZGZ02], les auteurs proposent une approche appelée "Motion Activity Map (MAM)", elle consiste en une génération d'une image synthétisée qui accumule l'activité du mouvement sur les grilles de l'image sur l'axe du temps, cette image représente les informations du mouvement et conduit à une représentation plus intuitive et plus compacte du mouvement.

D'autres descripteurs de mouvement ont également été proposés dans la littérature : descripteur d'activité de mouvement (Motion Activity), descripteur du mouvement de caméra (Camera Motion), descripteur WPD (Warping Parameter Descriptor), descripteur du mouvement paramétrique (Parametric Motion).

Inspirés de [FHY09], [VECR10] calculent plusieurs descripteurs de mouvement basés sur l'analyse du flux optique. Ils supposent que chaque locuteur possède ses propres gestuelles et expressions qui, décrites avec les bons attributs, peuvent être très discriminantes (par exemple le mouvement des mains souvent visible dans la région d'intérêt du costume). Pour caractériser ces particularités de façon robuste et efficace, ils proposent de déduire du flux optique des descripteurs de mouvement pour l'image globale, ainsi que pour les régions d'intérêt du visage et de la poitrine (qui est la même que celle du costume). Les amplitudes et orientations de vitesse et d'accélération sont calculées comme les dérivées première et seconde des points d'intérêts de l'image globale et des régions d'intérêts, celles-ci étant évaluées comme les coordonnées r et θ dans un système polaire. Ils proposent également de calculer l'amplitude relative présentée comme le rapport des amplitudes pour les régions d'intérêts du visage et de la poitrine sur celle de l'image globale.

2.4.5 Discussion

Le choix du descripteur adapté est dépendant de l'application. On retiendra que les meilleurs résultats sont logiquement obtenus par la combinaison de l'information de couleur avec des informations supplémentaires au prix d'une notable augmentation du temps de calcul et des dimensions du descripteur. Les histogrammes et les moments gardent un intérêt grâce à leur rapidité d'acquisition ainsi qu'à leur compacité. Les descripteurs basés sur les points d'intérêts s'avèrent plus efficace pour la détection d'objets qui sont généralement caractérisés visuellement par l'apparition de coins. On notera aussi que plus le nombre de descripteurs considérés augmente plus la performance du système d'indexation est meilleure. Il serait tout de même préférable de fusionner le résultats obtenus par des descripteurs de différentes modalités. Dans ce contexte et pour une raison purement

stratégique liée à la facilité de mise en œuvre, on privilège généralement une fusion tardive des descripteurs (voir la section 2.8).

2.5 Agrégation des descripteurs locaux

2.5.1 Sacs de mots visuels

La représentation en sac de mots est une méthode qui consiste à représenter un document par l'ensemble de mots qui le constituent et qui appartiennent à un dictionnaire prédéfini de mots, connu aussi sous les noms de "codebook' ou "vocabulaire". C'est une technique très réputée et largement utilisée dans le domaine de la recherche textuelle. Dans le cas des images et vidéos on parle de sac de mots visuels. La représentation en "Sacs mots de mots visuels" a été introduite dans le domaine de la recherche de vidéos par Sivic [SZ03]. Le vocabulaire est construit en utilisant un ensemble d'apprentissage Dev. Des caractéristiques visuels sont extraites des éléments de l'ensemble Dev, il en résulte un très grand nombre de points d'intérêts. Ce grand nombre est important pour la robustesse de la classification, mais évoque un défit de représentation à cause de la grande dimensionalité (SIFT est d'une dimension égale à 128). Un regroupement (Clustering) est appliqué sur l'ensemble des points d'intérêts résultants, comme par exemple, la méthode "K-means" qui est très utilisée dans ce contexte. On obtient donc un ensemble de groupes (clusters), le centre de chacun d'entre eux représente un mot visuel. L'ensemble de ces mots visuels constitue le "vocabulaire" ou le "codebook". La représentation final d'une image est l'histogramme, c'est à dire, la fréquences des mots visuels dans cette image. Cela est réalisé en faisant une correspondance entre les éléments du dictionnaire dont on dispose et chaque caractéristique extraite de image requête (image-clé dans le cas de vidéos) pour sélectionner le mot visuel le plus similaire à chaque caractéristique, et compter ensuite les fréquences. On pourrait aussi utiliser d'autres mesures au lieu de la fréquence.

Comme décrit dans [LP05], ce modèle est très efficace pour l'indexation des images. Cependant, l'absence de relations spatiales et les informations de localisation des mots visuels sont les principaux inconvénients de cette représentation. Ce modèle est utilisé dans l'état de l'art dans des approches non-supervisées pour l'extraction des thèmes/sujets cohérents en utilisant l'analyse sémantique latente [MGP03] et l'analyse sémantique latente probabiliste [MGP04]. D'autres approches supervisées ou semi-supervisées discriminatives utilisent cette méthode de représentation, comme par exemple, SVM et/ou des classificateurs bayésiens et/ou le modèle de langue [LJ06, ZLS07, TCG08].

2.5.2 Noyaux de Fisher

Les noyaux de Fisher ont été introduits pour combiner les avantages des approches génératives et discriminantes [JH99], afin d'accroître les performances de classification. Autrement dit, un noyau de Fisher fournit un moyen pour extraire des caractéristiques discriminantes à partir d'un modèle génératif. Jaakkola et

Haussler [JH99] ont prouvé qu'un classificateur linéaire basé sur un noyau de Fisher est, au moins, aussi performant qu'un modèle génératif.

Soit un modèle génératif $P(d|\Theta)$, Jaakkola propose de calculer le score de Fisher du document d :

$$U_d = \underset{\Theta}{\nabla} logP(d|\Theta) \tag{2.4}$$

Où l'opérateur $\underset{\Theta}{\nabla}$ représente le gradient par rapport à Θ. U_d est alors un vecteur dont la dimension est égale au cardinal de Θ. En ce sens, U_d est une représentation vectorielle du document d par rapport à un modèle génératif de paramètres Θ. Chaque composante du vecteur représente combien le paramètre du modèle génératif contribue à générer l'exemple donné. à l'aide de ce score, on peut alors définir une similarité entre deux exemples d_1 et d_2 à l'aide du noyau de Fisher suivant :

$$K(d_1, d2) = U_{d_1}^T I^{-1} U_{d_2} \tag{2.5}$$

où I est appelée la matrice d'information de Fisher.

Cette fonction de noyau est appelée noyau de Fisher. Elle peut être utilisée avec n'importe quel classificateur à base de noyau (e.g. SVM). La matrice I est habituellement approchée à la matrice identité, ce qui simplifie la fonction noyau :

$$K(d_1, d_2) = U_{d_1}^T U_{d_2} \tag{2.6}$$

Ainsi, la fonction noyau est simplifiée en un produit scalaire entre les vecteurs représentatifs des deux documents, ce qui peut être vu aussi comme un noyau linéaire entre ces deux vecteurs.

De nombreux travaux ont montré l'efficacité de l'utilisation des noyaux de Fisher [JH99, JDH00, SGN01, FNG01, VG01, PD07b]. Dans [PD07b], les auteurs ont appliqué le noyau de Fisher sur un vocabulaire visuel dans le cadre de la classification des images, où les mots visuels ont été modélisés par un modèle de mélange de gaussiennes (GMM). La représentation par noyau de Fisher tend à étendre la représentation par "sac de mots visuels" (BoVW). En effet la première représentation n'est pas limitée au nombre d'occurrences de chaque mot visuel, et comparée à la représentation BoVW, moins de mots visuels sont requis par cette représentation plus sophistiquée.

2.6 Optimisation des descripteurs

Avant d'utiliser un descripteur dans un système d'apprentissage ou de classification, il est recommandé de le soumettre à une chaine de pré-traitements, cette phase est appelée : "optimisation de descripteurs". L'optimisation des descripteurs passe par deux étapes importantes : la normalisation des descripteurs et la réduction de leurs dimensionnalités. La normalisation des descripteurs tend à modifier les valeurs des différentes composantes du vecteur caractéristique. La réduction de dimensionnalité quant à elle, et comme son nom l'indique est une méthode permettant de réduire le nombre de composantes formant le descripteur. Un descripteur optimisé a tendance a être plus efficace dans une approche de

classification. Il est aussi possible d'appliquer une seule des deux méthodes avant d'utiliser le descripteur (i.e. ou bien normaliser le descripteur ou bien réduire sa dimensionalité).

2.6.1 Normalisation des descripteurs

On peut distinguer deux catégories de méthodes de normalisation : normalisation par rapport à l'amplitude et normalisation par rapport à l'échelle. La première catégorie d'approche consiste à uniformiser la distribution des valeurs, de façon à ce qu'il n'y ait pas un grand écart entre les différentes valeurs. Cela s'avère très utile pour réduire l'influence les grandes valeurs qui dominent les petites valeurs. La deuxième catégorie tend à étaler l'ensemble de valeurs, de manière à ce qu'elles couvrent le maximum possible d'un intervalle donné. Autrement dit, ces approches font en sorte que la partie non nulle s'étale sur toute l'échelle ou l'intervalle. Cela est similaire à l'étirement d'un histogramme.

Soit X un ensemble de N vecteurs caractéristiques d'un ensemble de données à normaliser, chaque vecteur x_i est composé de d dimensions : $x_i = (v_1; v_2; :::; v_d)$. Nous proposons dans ce qui suit, quelques techniques de normalisation, qui sont souvent utilisées dans pour la représentation des images et vidéos.

- Normalisation L_1 et L_2 :

$$x\prime_{ij} = \frac{x_{ij}}{\|x_i\|} \quad où : \; j = 1, ...d \tag{2.7}$$

  où x_{ij} correspond à la valeur de la $j^{ème}$ composante du vecteur x_i, et $\|.\|$ dénote la norme d'un vecteur, qui dans $L1$ est égale à : $\sum_j x_{ij}$ et dans L_2 elle est égale à $\sum_j x_{ij}^2$.
- Min-Max :

$$x'_{ij} = l + \frac{(u-l) \times (x_{ij} - min_j)}{max_j - min_j} \tag{2.8}$$

  où x_{ij} est la $j^{ème}$ composante du vecteur x_i, min_j et max_j correspondent respectivement aux valeurs minimale et maximale de la $j^{ème}$ composante dans X (la $j^{ème}$ composante des différents vecteurs $x_i \in X$). u et l correspondent aux extrémités du nouvel espace (intervalle cible). Généralement la normalisation est réalisée de façon à projeter les valeur du vecteur résultant $x_i\prime$ dans l'intervalle $[0, 1]$
- σ-norm : consiste à centrer-normer les valeurs de chaque bin.

$$\sigma_j = \frac{\sum_{i=1}^{N}(x_{ij} - \bar{x_{ij}})^2}{N}; \quad j = 1, \ldots, d \tag{2.9}$$

$$\acute{x_{ij}} = \frac{x_{ij} - \bar{x_j}}{\sigma_j} \quad i = 1, \ldots, n \; et \quad j = 1, \ldots, d \tag{2.10}$$

  où d indique la taille du vecteur x_i (i.e. le nombre de ses composantes), n est le nombre d'échantillons (vecteurs) dans la collection ; $\bar{x_i}$ et σ_i sont les valeurs de la moyenne et la variance du $i^{ème}$ *bin* du vecteur, respectivement.

2.6.1.1 Transformation de loi de puissance (Power-law)

Le but de la transformation de loi de puissance est de normaliser les distributions des valeurs, en particulier dans le cas de composantes d'un histogramme. Elle consiste simplement en l'application d'une fonction de transformation f sur toutes les composantes du vecteurs individuellement [SDH+11] :

$$f(x) = sign(x) \times |x|^{\alpha} \tag{2.11}$$

où sign(x) est le signe de x La transformation "Power-law" a été appliquée par Jégou et al. dans [JPD+11], où les auteurs ont appliqué cette normalisation sur un descripteur basé sur le *noyau de Fisher*. Ils ont constaté empiriquement, que cette étape de normalisation améliore constamment la qualité de la représentation. Ils ont donné plusieurs interprétations complémentaires qui justifient cette transformation.

2.6.2 Réduction de dimensionnalité

La réduction de la dimensionnalité est une technique, qui comme son nom l'indique, vise à réduire la dimension des données par leur projection dans un autre espace de dimension inférieure, sans écarter de l'information significative, ou pour être plus précis en gardant le maximum possible d'informations, car cette projection va causer une perte d'information, dépendamment du nombre et du choix des dimensions éliminées. L'objectif de la réduction de dimensionalité est de trouver à partir d'une combinaison (linéaire ou non) des dimensions initiales du vecteur en question un nouvel espace de dimension significativement inférieure qui contient une grande part de l'information totale, dans le but de trouver une représentation discriminative des données. Cela permet d'une part, de s'affranchir du fléau de la dimension, et d'autre part, ça permet aussi de débruiter les données. Nous présentons dans ce qui suit quelques approches qui sont largement utilisées pour réduire la dimentionnalité des descripteurs dans les système d'indexation multimédia.

2.6.2.1 L'analyse en composantes principales (ACP)

L'analyse en composantes principales (Principal Component Analysis (PCA) en anglais) est une transformation linéaire orthogonale qui projette les données dans un espace de dimensions inférieures ou égales d'attributs non corrélées appelées composantes principales (PC), d'où le nom : "Analyse en composantes principales". Elle utilise la variance (énergie) comme une mesure de l'information précieuse et dérive les nouvelles données de manière à maintenir le maximum d'information possible. L'ACP prend sa source dans l'article [Pea01]. Soit $(X_1, X_2, \ldots, X_p)$ les composantes initiales d'un espace p, Le but de l'ACP est de trouver de nouvelles composantes$(C_1, C_2, \ldots, C_k)$. Les C_k sont issues d'une combinaison linéaire de l'ensemble des composantes X_i de p :

$$C_k = \alpha_{1k}X_1 + \alpha_{2k}X_2 + \cdots + \alpha_{pk}X_p \tag{2.12}$$

Les composantes C_k ne sont pas corrélées deux à deux, ont une variance maximale et ordonnées par ordre décroissant par rapport à la variance.

Dans le domaine de la recherche d'informations, une technique populaire pour effectuer cette transformation consiste en l'application d'une décomposition en valeurs singulières (SVD) de la matrice de variance-covariance, en utilisant la décomposition des vecteurs propres. SVD décompose une matrice A de $m \times n$ en un produit de trois matrices : T de dimensions $m \times r$, S de dimensions $r \times r$, et D de dimensions $r \times n$:

$$A = TSD^t \tag{2.13}$$

tel que $TT^t = DD^t = D^tD = I$, où I est la matrice identité. S est une matrice diagonale, dans laquelle les éléments de la diagonale sont appelés valeurs singulières de la matrice A, et sont dans un ordre décroissant de façon monotone. D^t dénote la transposée de la matrice D. Il est prouvé que les k plus grandes valeurs singulières ensemble, en plus des vecteurs propres qui leur correspondent, encodent l'information la plus important de la matrice A [DDF+90]. Par conséquent, A est souvent approximée à A^* (i.e. $A = A^*$) :

$$A^* = T_k S_k D_k^T \tag{2.14}$$

Ainsi, les axes de la plus grande variance correspondent aux vecteurs propres associés aux plus grandes valeurs propres. La nouvelle base de dimension réduite est enfin formée par les k-vecteurs propres correspondant aux k-plus grandes valeurs propres. Il n'existe aucune méthode pour sélectionner automatiquement le nombre k ; généralement, il est décidé a priori ou il est sélectionné par seuillage des valeurs propres. La suppresion des vecteurs de faible variance a un avantage supplémentaire. En effet, ces vecteurs sont connus pour être bruités. En les supprimant on ne réduit pas non seulement la dimension du descripteur mais aussi le bruit qu'il contient.

2.6.2.2 L'analyse en composantes indépendantes (ACI)

L'analyse en composantes indépendantes (Independent Component Analysis (ICA) en anglais) a été introduite par Jutten et Herault [JH91] comme une méthode de séparation aveugle de sources, mais ce n'est que dix ans plus tard, grâce au travail de Hyvärinen et Oja [HO00] que l'ICA a commencé à recevoir de l'attention nécessaire dans de nombreux domaines de traitement de signal, et plus particulièrement dans l'analyse spectrale de l'image pour des tâches telles que la classification [MVB+11], la réduction de la dimensionnalité [WC06, MQ12] , déconvolution spectrale [XLWZ11] ou la détection de cibles [TAS11]. Le but de cette technique est de trouver une représentation linéaire des données non gaussiennes de telle sorte que les composantes sont aussi indépendantes que possible.

2.6.2.3 L'analyse discriminante linéaire (LDA)

L'analyse discriminante linéaire (LDA) est une méthode permettant de trouver une combinaison linéaire de variables qui sépare mieux deux ou plusieurs classes. LDA n'est pas un algorithme de classification, bien qu'elle utilise des

étiquettes des classes. Toutefois, le résultat de LDA est principalement utilisé dans le cadre d'un classificateur linéaire. D'autre part, une utilisation alternative apporte une réduction de la dimension avant d'utiliser des algorithmes de classification non linéaires. La réduction de dimensionnalité par LDA améliore souvent la performance des classificateur. Dans ce contexte, Song et al. ont montré que la performance de leur système de classification a une meilleure précision après l'analyse des données par LDA que PCA [MSY05].

2.7 Apprentissage automatique/Classification

L'apprentissage automatique est appliqué dans les cas où un programmeur ne peut pas indiquer explicitement à la machine ce qu'il faut faire et quelles sont les mesures à prendre. De nos jours, et depuis plusieurs années, plusieurs applications tirent profit de l'apprentissage automatique. L'exemple typique de ce genre d'application est celui dans lequel on a besoin de retourner une liste de résultats ordonnés en termes de pertinence par rapport à une requête donnée. Parmi les plus connues et plus utilisées par la majorité des utilisateurs, on peut en citer les applications de traduction automatique, la reconnaissance de visages qui peut faire partie de systèmes de sécurité ou d'accès contrôlés à des endroits ou services, la reconnaissance vocale, la reconnaissance des empreintes digitales, etc. L'objectif de l'apprentissage automatique est d'apprendre à partir d'un ensemble de données dit " ensemble d'apprentissage", des informations de manière à pouvoir les obtenir ces mêmes types d'informations sur des données non encore vues.

L'apprentissage automatique peut se heurter à plusieurs problèmes, parmi lesquels : le sur-apprentissage (Overfitting en anglais). Le sur-apprentissage est une erreur de modélisation qui se produit quand le modèle apprend une fonction trop conforme à un ensemble limité de données, et qui aura donc, du mal à la généraliser face à de nouveaux échantillons. En réalité, les données étudiées ont souvent un certain degré d'erreur ou de bruit aléatoire. Ainsi la tentative de rendre le modèle conforme de trop près à une petite quantité de données inexactes peut infecter le modèle avec des erreurs substantielles et de réduire son pouvoir prédictif (pouvoir de généralisation).

Pour estimer la fiabilité du modèle appris, pour optimiser les paramètres d'un modèle et/ou pour éviter le problème du sur-apprentissage, on utilise une méthode appelée "validation croisée" (cross-validation en anglais), qui consiste à diviser l'ensemble d'apprentissage en deux parties : une pour bâtir le modèle et la seconde pour l'évaluer. Il y a différentes manières d'opération possibles :

- Diviser l'ensemble d'apprentissage en deux parties, typiquement $> 60\%$ des échantillons pour la génération du modèle et le reste pour le test. L'erreur ou une mesure de performance est estimée en fonction du problème étudié (e.g. La précision, l'erreur quadratique moyenne).
- Diviser l'ensemble d'apprentissage en k parties, pour apprendre le modèle sur l'un des $(k-1)$ ensembles et l'évaluer sur l'ensemble restant. L'opération est répétée k fois, en sélectionnant à chaque itération un ensemble de validation non pris en compte dans les itérations précédentes. Cette procédure est

appelée “K-fold cross-validation”. Les mesures de performance ou les erreurs calculées au fur des itérations sont moyennées pour calculer une mesure de performance/l’erreur finale.

La validation croisée est très souvent utilisée pour l’optimisation des paramètres des méthodes d’apprentissage [SWHO11] et aussi pour éviter le problème du sur-apprentissage [Ste07].

2.7.1 Approches supervisées vs. Non supervisées

On distingue deux classes d’apprentissage automatique : l’apprentissage supervisé et l’apprentissage non supervisé, souvent référencé par le terme anglais : “clustering”. Dans l’apprentissage supervisé, l’apprenant vise à sélectionner la meilleure fonction $g : X \rightarrow Y$ permettant de faire correspondre un ensemble de m données x_i décrites dans un espace X à des classes ou des catégories cibles $y_i \in Y$. Le but est de choisir la fonction la plus précise possible répondant à un ou plusieurs critères d’optimisation, tout en gardant un pouvoir de généralisation pour les exemples non encore vus. Dans l’apprentissage supervisé, l’apprenant dispose d’un l’ensemble d’apprentissage $D_{train} = \{(x_i, y_i)_{i=1}^m\}$ où $x_i \in X$ et $y_i \in Y$. Autrement dit, chaque élément de l’ensemble d’apprentissage est annoté. L’apprenant va utiliser donc la représentation de chaque échantillon ainsi que son annotation pour apprendre une fonction g et de la généraliser pour des données non encore vues (i.g. non prises en compte lors du processus d’apprentissage) dites : “données de test”. Si les étiquettes ou les classes sont de type continu (c’est-à-dire des valeurs réelles), on parle de “régression”.

Dans l’apprentissage non supervisé, les classes des échantillons d’apprentissage sont inconnus. L’ensemble d’apprentissage n’est constitué donc que des descriptions des échantillons d’apprentissage : $D_{train} = \{(x_i \in X)_{i=1}^m\}$. L’apprenant tente sur la base de certaines mesures, typiquement des distances, d’étudier l’existence de regroupements d’exemples. Les exemples de chaque groupe (cluster en anglais) sont censés avoir des caractéristiques en commun. La méthode des *K-means* est l’une des méthodes de clustering les plus connues.

Il existe une autre classe d’apprentissage automatique dite semi-supervisée, qui est en quelque sorte un type intermédiaire entre les méthodes supervisée et non-supervisée. L’apprentissage semi-supervisé fait usage d’un ensemble de données qui ne sont pas toutes annotées, typiquement un petit nombre de données annotées et une grande quantité de données non étiquetées : $D_{train} = \{(x_i, y_i)_{i=1}^{m\prime}\} \cup \{(x_i)_{i=m\prime+1}^m\}$ où $1 < m\prime < m$.

Dans [ASS01], les auteurs montrent qu’il est possible de transformer un problème de classification multi-classes en plusieurs problèmes bi-classes en utilisant le principe de “un contre tous” (one-vs-all). Chaque système binaire classifie les échantillons dans une classe ou dans une autre qui comprend toutes les classes restantes. Il existe une autre stratégie “un contre un” (one-vs-one), qui consiste à générer un classificateur par chaque paire de classes. La classe qui reçoit le plus de votes est attribuée à l’exemple en question.

2.7.2 Approches génératives

Étant donné un ensemble C de m classes et un ensemble d'échantillons X, les approches génératives modélisent la probabilité jointe $p(x, c_i)$ d'un échantillon $x \in X$ et une classe $c_i \in C$ et prédisent la classe à la quelle x est le plus à même d'être classé. Pour ce faire, elles s'appuient sur le calcul des probabilités $P(c_i|x)$ en utilisant le théorème de Bayes :

$$P(c_i|x) = \frac{P(x|c_i) * P(c_i)}{P(x)} \tag{2.15}$$

où :
- $P(c_i|x)$: La probabilité a posteriori de c_i sachant x ;
- $P(c_i)$: la probabilité a priori de c_i, aussi appelée la probabilité marginale de c_i ;
- $P(x)$: la probabilité a priori ou marginale de x ;
- $P((x|c_i)$: Probabilité de x sachant c_i, avec c_i paramètre connu (fixe). Elle est aussi connue sous les noms de : 1) fonction de vraisemblance de c_i, ou 2) la densité de probabilité de la classe c_i

Dans le cas où les probabilités a priori sont égales pour les différentes classes, la décision peut être réalisée en se basant uniquement sur les fonctions de vraisemblance $P(x|C_i)$ de chaque classe.

Une méthode générative typique s'appuie sur un modèle de mélange gaussiennes (GMM) [Bishop 2007] pour modéliser la distribution des échantillons d'entrainement. L'ensemble des paramètres des GMM peut être efficacement appris en utilisant l'algorithme "espérance-maximisation", souvent abrégé EM [Del02].

Les méthodes bayésiennes sont particulièrement adaptées quand la dimension des données en entrée est petite. L'estimation des paramètres se fait en utilisant la méthode de maximum de vraisemblance [Ald97, Moo96]. La popularité de la classification bayésienne est accentuée dans le domaine de l'analyse et la recherche textuelles [IT95], spécialement pour la détection des SPAM [SDHH98] et la classification des emails [Pro99]. En plus de sa simplicité, cette méthode s'avère plus efficace dans les situations complexes du monde réel. Contrairement à plusieurs autres méthodes supervisées, la classification bayésienne nécessite peu de données d'apprentissage pour l'estimation des paramètres du modèle.

2.7.3 Approches discriminatives

Les méthodes discriminatives quant à elles, modélisent la probabilité a posteriori $P(c_i|x)$ directement, ou apprennent une correspondance directe entre les données en entrées et les différentes classes cibles. Il existe plusieurs raisons impérieuses et convaincantes poussant à favoriser l'utilisation des méthodes discriminatives au détriments des méthodes génératives, une d'entre elles, est succinctement articulée par Vapnik [Vap98], qui dit qu'on a généralement tendance à résoudre le problème de classification directement et qu'on ne se préoccupe pas d'un problème plus général comme étape intermédiaire (comme modéliser

$P(x|c_i)$ par exemple). Indépendamment des problèmes de calculs et de l'absence de données, le consensus qui régnant stipule que les méthodes discriminantes sont généralement préférée aux méthodes génératives [NJ01].

2.7.3.1 K-plus proches voisins

K- Plus proches voisins (*K-Nearest Neighbors* en anglais, souvent abrégé en K-NN) est une méthode bien connue dans le domaine d'apprentissage et la vision [SDI08]. Contrairement à certaines approches, la phase d'apprentissages dans K-NN est transparente. En effet, K-NN nécessite une mémorisation de l'ensemble des données d'apprentissage parce qu'elle ne fait aucune généralisation. Elle consiste à classer un objet par vote majoritaire de ses voisins. La classe la plus dominante parmi ses K- plus proches voisins (K est généralement de petite taille) lui sera attribuée. Si $K = 1$, l'objet est simplement affecté à la classe de son voisin le plus proche ; son nom se voir simplifié dans ce cas à "the nearest neighbor" (1-NN). Cette méthode présente plusieurs inconvénients. D'une part, elle est gourmande en temps et mémoire parce qu'elle nécessite le chargement de l'ensemble des données d'apprentissage, et le calcul des distances entre chaque exemple test et l'ensemble des exemples d'entrainement. Cela augmente le temps de calcul. K-NN est aussi sensible à la présence de données bruitées, ce qui la rend difficile à généraliser. Une solution possible consiste à choisir un sous ensemble de vecteurs caractéristiques non bruités [BL97]. Cette méthode se heurte à un autre problème lorsque certaines classes sont représentées par peu d'individus. En effet, les classes les plus fréquentes ont tendance à dominer les classes rares dans la prédiction. À cause de leur grand nombre, les classes dominantes seront plus représentées dans les K- plus proches voisins, et par conséquent, elles affecteront le vote majoritaire. Pour remédier à ce problème, une version de cette méthode consiste à pondérer le vote de chacun des K plus proches voisins par la distance le séparant de l'exemple test à classer. Il est recommandé de comparer les résultats des nouveaux algorithmes d'apprentissage à ceux de 1-NN parce que les performances de cette dernière approches sont constantes et souvent bonnes [JDM00].

2.7.3.2 Approches à noyaux

Les méthodes à noyaux sont surtout sollicitées dans le cas de données non linéairement séparables. Ce genre d'approches se basent sur le théorème de "Mercer" [Sch02], qui dit que chaque fonction noyau continue symétrique semi-définie positive peut être exprimée en un produit scalaire dans un espace de haute dimension. La première application des noyaux dans l'apprentissage automatique remonte à l'année 1964 avec le travail de Aizerman et al, où les auteurs présente une version à base de noyau de l'algorithme perceptron de "Rosenblatt". En effet, il n'y a que récemment que les chercheurs ont reconnu la grande importance et la large applicabilité des noyaux. Avec le travail de Boser et al. [BGV92], SVM "machines à vecteurs de supports", devient la plus illustre technique à base de noyau. Nous présentons cette méthode dans la section 2.7.3.2. Pour séparer des données non linéairement séparables, ces méthodes simulent le passage dans un

espace de grande dimension. Dans cet espace qu'il n'est pas nécessaire de manipuler explicitement, les méthodes linéaires peuvent être mises en œuvre pour y trouver des régularités linéaires, correspondant à des régularités non linaires dans l'espace d'origine.

Grâce à l'utilisation des fonctions noyau, il devient ainsi possible d'avoir le meilleur des deux mondes : utiliser des techniques simples et rigoureusement garanties, et traiter des problèmes non linéaires. C'est pourquoi ces méthodes sont devenues très populaires récemment.

Machines à vecteurs de supports

Machines à vecteurs de supports (SVM) est une des méthodes les plus populaires dans la famille des approches discriminatives, et de méthodes à base de noyau, de classification. Elle fut développée par Vapnik [CV95] en 1995, et demeure à ce jour un des algorithmes les plus utilisés, spécialement pour la reconnaissance de formes. Cela est dû d'une part, à sa capacité de généralisation et d'autre part, à la notion de noyau qui la rend mieux adaptée pour résoudre le problème des données non linéairement séparables. Grâce à ces qualités, cette méthode a été adoptée par la communauté des images et vidéos, où les vecteurs caractéristiques de ces données sont de grandes dimensions. [CHV99] est une des plus récentes recherches visant à appliquer SVM pour la classification des images. Étant données deux classes de données de dimension d, le principe de base de SVM est de trouver un(des) hyperplan(s) séparateur(s) distinguant parfaitement les deux classes de données, en maximisant la marge séparant les données de cet hyperplan. La figure 2.4 illustre ce principe dans le cas d'un espace à deux dimensions. H représente l'hyperplan séparateur séparant les cercles noirs des blancs ; quant aux exemples sur les marges sont appelées "vecteurs de support". Comme les données ne sont généralement pas linéairement séparables, SVM utilise la notion de noyau permettant de projeter les données dans un espace de haute dimension où les données seront linéairement séparables [SS01]. L'apprentissage des paramètres de l'hyperplan séparateur et des marges se fait en utilisant les exemples de développement. SVM vise à trouver l'hyperplan optimal maximisant la marge entre l'hyperplan et les vecteurs de support.

Il existe plusieurs types de noyau :

- Linéaire (simple produit scalaire) :

$$K(x_i, x_j) = x_i \dot{x}_j \tag{2.16}$$

- RBF (Radial Basis Function) :

$$K(x_i, xj) = exp(-\frac{||x_i - x_j||^2}{2 * \sigma^2}) \tag{2.17}$$

- Polynomial :

$$K(x_i, xj) = (x_i \dot{x}_j + c)^2 \tag{2.18}$$

- Sigmoïde :

$$K(x_i, xj) = tanh(x_i \dot{x}_j + c) \tag{2.19}$$

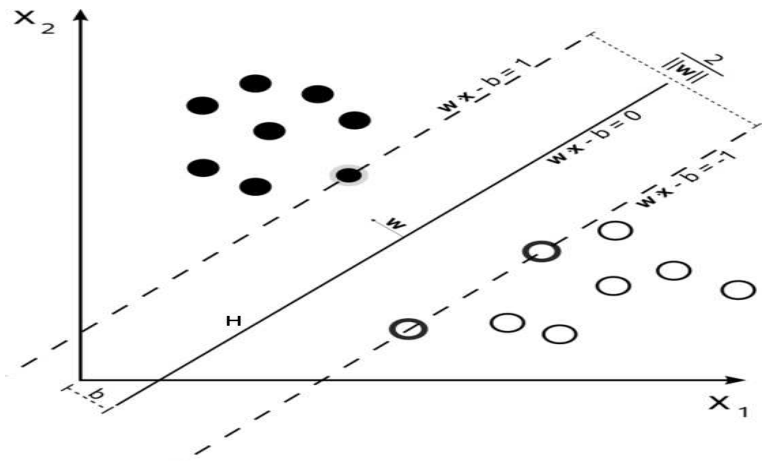


FIGURE 2.4 – Séparation linéaire dans un espace à deux dimensions.

avec $||.||$ la norme L2. x_i, xj sont deux vecteurs différents, et σ un paramètre gaussien à optimiser par cross-validation. Cela mène à une matrice symétrique appelée "matrice de noyau", qui indique la similarité entre chaque paire de vecteurs en entrée. Généralement, uniquement les fonctions de similarité qui conduisent à une matrice satisfaisant les conditions de Mercer, peuvent être utilisées.

Les noyaux RBF donnent généralement les meilleurs résultats, comme montré dans [LNSF02] pour le cas de la reconnaissance de chiffres manuscrits.

2.7.4 Apprentissage par modèle d'ensemble

Généralement, il n'existe pas d'algorithme d'apprentissage individuel qui dans n'importe quel domaine toujours induit le modèle le plus précis. En effet, chaque algorithme d'apprentissage se base sur un ensemble d'hypothèses, dans le cas où ces dernières ne conviennent pas aux données considérées, cela mène à des erreurs et des performances très basses en termes de précision. Pour remédier à ce problème, certains chercheurs ont proposé des méthodes qui utilisent non seulement un seul mais un ensemble d'algorithmes d'apprentissage, qui peuvent être de même type ou de types différents. Ces techniques sont appelées : "techniques d'apprentissage par ensemble". L'idée de ce type de méthodes est de construire un groupe d'apprenants de base qui, une fois combinés, génèrent un "méta-modèle" qui a une meilleure précision que les apprenants individuels. Les apprenants de base ne sont pas sélectionnés en matière de performance, mais en termes de simplicité. Plusieurs méthodes d'apprentissage à base d'ensembles ont vu le jour, on peut les catégoriser en quatre grandes familles : "voting", "bagging", "boosting", "stacking". Ces méthodes diffèrent au niveau de la façon avec laquelle les décisions des apprenants de base sont combinés. Pour plus de détails sur les méthode d'apprentissage par ensemble on peut se référer au livre de Zhaou [Zho12].

2.7.4.1 Voting

Cette stratégie consiste à combiner les décisions (prédictions) de l'ensemble des apprenants individuels :

$$y = f(d_1, d_2, ..., d_i, .., d_n|\phi)$$

où :
n : est le nombre d'apprenants considérés.
y : est la prédiction finale.
ϕ est l'ensemble des paramètres
d_j est la décision du j^{eme} apprenant.
f est la méthode du vote.

La plus simple technique de vote est le *vote majoritaire*. On peut aussi pondérer les votes des apprenants individuels par des degrés de confiance ou d'importance (poids). Cette méthode est appelée "vote pondéré" ("weighted voting" en anglais).

$$y = f(d_1, d_2, ..., d_i, .., d_n|\phi) = \sum_{j=1}^{n} d_j.w_j \tag{2.20}$$

avec : $w_j \geq 0$ et $\sum_{j=1}^{n} w_j = 1$
où w_j est le poids accordé au j^{eme} apprenant.
La valeur des poids peut être reliée par exemple à la performance des apprenants.

2.7.4.2 Bagging

Bagging est l'abréviation de "Boostrap aggregating ". Le Bagging a été proposé par Breiman [Bre94] afin d'améliorer la performance de classification en combinant les classification d'un ensembles de données générées aléatoirement. C'est une méthode de type "voting" dans laquelle, les apprenants sont entraînés sur des ensembles de données légèrement différents. En effet, pour chaque apprenant A, un nouvel ensemble de données d'apprentissage est généré par tirage aléatoire avec remise (boostrap) des exemple depuis l'ensemble de données d'apprentissage d'origine. Les décisions des apprenants de base sont ensuite combinées suivant un vote majoritaire. N'importe quel type de modèle de classification peut être utilisé comme apprenant de base. La technique de bagging est typiquement utilisée pour les algorithmes d'apprentissage dits : "instables", pour lesquels, un petit changement dans les données d'apprentissage peut causer un changement signifiant dans le modèle résultant. Les réseaux de neurones et les arbres de décisions sont de bons candidats pouvant bénéficier des avantages du Bagging. Dans [SQ10], Safadi et al. propose un schéma de Bagging basé sur les SVM avec une sélection biaisée des échantillons positifs et négatifs pour traiter le problème des classes déséquilibrées dans le cadre de l'indexation des documents multimédia.

multi-SVMs : Safadi et al. [SQ10] proposent une méthode qui consiste à combiner m classifieurs via une stratégie de "Bagging" où chacun d'entre eux utilise tous les échantillons d'apprentissage de la classe dominée (typiquement,

la classe positive) et un ensemble d'échantillons de la classe dominante (typiquement, la classe négative) est tiré aléatoirement avec remise (boostrap), avec :

$$m = (f_{neg} \times N_{neg})/(f_{pos} \times N_{pos}) \tag{2.21}$$

où N_{pos} est le nombre d'exemples positifs, N_{neg} est le nombre d'échantillons négatifs, f_{neg} et f_{pos} sont des paramètres (entiers positifs non nuls) relatifs aux classes positive et négative, respectivement. Nous rappelons à ce stade, que l'annotation concerne une paire de concepts et non un concept individuel. f_{pos} gère la proportion des échantillons de la classe dominante qu'on veut utiliser, par rapport au nombre d'exemples de la classe dominée (e.g. Deux fois plus d'exemples négatifs que positifs). f_{neg} quant à lui, permet de contrôler, à l'aide de f_{pos}, le nombre de classifieurs souhaité. L'ensemble E_D est divisé en m sous-ensembles, où chaque sous-ensemble contient tous les exemples positifs contenus dans E_D et $(f_{pos} \times N_{pos})$ exemples négatifs sont tirés aléatoirement avec remise. Ensuite chacun des m classifieurs est entrainé sur un sous-ensemble différent. On remarque que la contrainte $f_{neg} \times N_{neg} \geq f_{pos} \times N_{pos}$ doit être vérifiée. Finalement, les scores des m classifieurs sont fusionnés en utilisant n'importe quelle fonction possible, typiquement une moyenne. Plus la valeur de m est grande, meilleure est la performance finale. Il a été montré qu'utiliser SVM comme classificateur de base donne les meilleurs résultats dans le domaine de l'indexation des vidéos. Nous appellerons par la suite, cette méthode globale de classification : "MSVM"'.

2.7.4.3 Boosting

La technique du boosting consiste à améliorer des apprenants ayant une performance faible mais supérieure à celle du hasard. Il a été prouvé dans [Sch90] qu'il est possible de transformer de tels apprenants en bons apprenants pouvant bien classer des exemples non annotés. La similarité entre le bagging et le boosting se résume uniquement dans la construction d'un ensemble de classificateurs en échantillonnant l'ensemble de données d'apprentissage, et en combinant les décisions des différents classificateurs suivant un vote majoritaire. L'échantillonnage des sous-ensembles d'apprentissage se fait dans le boosting de manière à fournir au prochain apprenant un ensemble de données le plus informatif possible. Parmi les méthodes de boosting les plus réputées, on peut citer "Adaboost" proposée par Freund et Schapire [FS97]. Cet algorithme accorde des poids à l'ensemble des exemples d'apprentissage. A chaque itération i, un classificateur C_i est entrainé de manière à minimiser l'erreur de classification. Cette erreur est calculée et utilisée par C_i pour mettre à jour la distribution des poids des exemples d'apprentissage. Cette mise à jour des poids des exemples est faite de manière à ce que les exemples mal-classés dans l'itération courante aient plus de chances d'être introduits dans l'ensemble d'apprentissage du classificateur de la prochaine itération. La processus est itéré jusqu'à ce qu'un critère d'arrêt, généralement lié au taux d'erreurs, soit vérifié. Plusieurs généralisations de AdaBoost au cas multi-classes ont vu le jour [ZRZH05].

2.7.4.4 Stacking

Le stacking est une technique d'apprentissage par ensemble qui ressemble beaucoup à la méthode de "voting", elle a été proposée par Wolpert [Wol92]. Le stacking consiste à combiner plusieurs classificateurs. Premièrement, un certain nombre de classificateurs sont entrainés sur l'ensemble de données d'apprentissage, leurs sorties sont ensuite combinées en utilisant un nouveau méta-classificateur pour apprendre un modèle permettant de faire une correspondance entre les décisions des classificateurs individuels de base et les classes correctes des exemples. Dans l'indexation multimédia, la technique de stacking est largement utilisée comme une méthode de fusion tardive, parce que c'est une bonne stratégie pour fusionner les scores des classificateurs, qui sont obtenus de différentes modalités. Une étude comparative entre les trois techniques : bagging, boosting et stacking est présentée dans [GLTT10].

2.7.5 Apprentissage profond (Deep learning)

L'apprentissage profond (Deep learning en anglais) et spécialement les réseaux de neurones à convolution (CNN) ont attiré beaucoup d'intérêts surtout dans le domaine de la vision par ordinateur et notamment la classification d'images. L'apprentissage profond a été proposé initialement par G. E. Hinton en 2006 [HS06] pour la représentation de données (image, audio, texte, etc) en imitant le mécanisme d'abstraction multi-couches du cerveau humain. Ils ont décrit un système combinant l'apprentissage des caractéristiques et la classification dans un unique processus d'apprentissage. Les CNNs peuvent être utilisés pour l'apprentissage non supervisé des caractéristiques et peuvent servir donc, d'outils pour générer de descripteurs de bas niveau. En effet, avec leur architecture multi-couches hiérarchique, les CNNs sont capables d'apprendre et reconnaitre des motifs visuels directement à partir des pixels des images. Il peuvent être également utilisés comme des apprenants renvoyant en sortie les classes des échantillonnés tout en faisant un apprentissage combiné des caractéristiques et de la séparation entre les différentes classes. De plus, les CNNs sont bien connus pour leur adaptation au pré-traitement minimal et leur robustesse à la distorsion [LKF10, YKY+13].

Les réseaux de neurones profonds ont été largement étudiés et appliqués pour la classification des images [CMM+11, KSH12], la reconnaissance des actions humaines [KLY07], la reconnaissance des gestes de la main [NDC+11], l'analyse des scènes [PC14], etc ; et ont démontré leur efficacité en atteignant de bonnes performances comparables à celles des meilleures approches de l'état de l'art. Les CNNs ont été également utilisés pour la détection de concepts visuels dans les vidéos avec des travaux qui ont fait l'objet de participation à la tâche d'indexation sémantique de la compagne TRECVid (voir la section 2.13.2.3) et ont montré des résultats intéressants [YKY+13].

2.7.6 Normalisation des sorties de classificateurs

La fusion d'informations, et spécialement la fusion tardive (voir la section 2.8.2), s'avère très importante et utile dans le domaine d'indexation et de recherche d'information. Surtout dans les systèmes multimodaux, où le but est de combiner des informations issues de sources hétérogènes. Cette fusion permet d'avoir des décisions plus robustes et précises. Généralement, ces informations sont des probabilités ou des scores de classification. En effet, dans ce contexte, un classificateur estime une valeur qui reflète à quel point un échantillon est susceptible d'appartenir à une classe donnée. Cette valeur n'est pas nécessairement une probabilité, et sa nature dépend amplement de l'algorithme utilisé pour la calculer. Les scores renvoyés peuvent appartenir à n'importe quelle plage de valeurs. Autrement dit, les scores ne sont pas nécessairement compris entre 0 et 1, comme c'est le cas des approches calculant les probabilités. Ainsi, deux classificateurs de même type qui sont entrainés sur des données différentes ou utilisant des descripteurs différents (sur le même ensemble de données), peuvent renvoyer des scores qui ne sont pas homogènes : qui n'appartiennent pas nécessairement à une même plage de valeurs (e.g. Entre 0 et 0.3 pour l'un, et entre 0.6 et 1 pour l'autre).

Ces situations engendrent plusieurs problèmes ; on peut en citer quelques-uns qui sont décrits à travers les questions suivantes :

1. Comment comparer les résultats de deux classificateurs ? Si par exemple, un premier classificateur renvoie le score 0.3 et un autre renvoie la valeur 0.9, lequel faut-il croire ?
2. Que signifie un score égal à 0.6 ? et comment interpréter deux scores différents, sachant que si deux classificateurs L_1 et L_2 renvoient respectivement deux scores $sc_1 = 0.5$ et $sc_2 = 0.8$, de classification d'une exemple e dans une classe C, ne signifie pas nécessairement que L_2 a plus de confiance que L_1 de classer e dans C. En effet, puisque ce sont des scores et non pas des probabilités, il est possible que L_2 renvoie des scores appartenant à une plage plus proche de 1 (par exemple [0.77 , 0.98], alors que les sorties de L_1 couvrent une plage de valeurs bornée par une valeur proche de 0.5 ([0.05, 0.6] par exemple). Dans ce cas, c'est plutôt L_1 qui est plus confiant dans sa classification de e. Il est donc primordial de passer à la notion de probabilité quand on veut parler de ce genre de sémantique, puisqu'un score permet uniquement de classer et trier des exemples.
3. Comment évier la dominance des grandes valeurs ? En fusionnant des scores, les plus grandes valeurs ont tendance à dominer, voire écraser, les petites valeurs.

Nous avons vu précédemment qu'il est indispensable de normaliser les sorties des classificateurs afin de les homogénéiser avant de les fusionner. Dans la littérature, un autre terme est également utilisé pour désigner la normalisation : "calibration". Avant de parler des techniques de calibration des scores, il est important de se poser la question suivante : *qu'est ce qu'un score bien calibré ?* Pour simplifier, nous allons considérer dans ce qui suit, le cas de la classification binaire,

où le score renvoyé par le classificateur reflète la probabilité qu'un exemple x appartienne à la classe donnée (classé positif par rapport à la classe en question). Cette probabilité sera dénotée par $P(y = +|x)$.

Un bon score peut être défini comme une valeur variant entre 0 et 1 et qui soit une bonne estimation de $P(y = +|x)$. Dans la réalité, les probabilités conditionnelles ne sont pas connues, des chercheurs ont proposé donc un modèle graphique, appelé "diagramme de fiabilité" (*Reliability diagram* en anglais, DeGroot et Fienberg, 1982/1983), qui permet de visualiser et estimer le degré de calibration d'une distribution. En divisant l'intervalle [0, 1] en 10 bins de tailles égales, le diagramme de fiabilité consiste à dessiner pour chaque bin, un point ayant comme coordonnées la valeur moyenne du score de ce bin et le nombre (voire la fréquence) des échantillons positifs dont le score tombe dans ce bin. Une distribution est dite bien calibrée si son diagramme de fiabilité coïncide avec la diagonale (voir la figure 2.5). Il existe certaines méthodes qui produisent des scores bien calibrés, comme par exemple, la régression logistique, l'analyse discriminante, les arbres de décision, etc. D'autres méthodes en revanche, donnent des scores qui ne sont pas bien calibrés, comme par exemple, SVM, Naive Bayes, Boosting, etc. Nous allons présenter dans ce qui suit quelques méthodes de normalisation. Nous noterons s_i et $s_i\prime$, le score avant et après normalisation, respectivement.

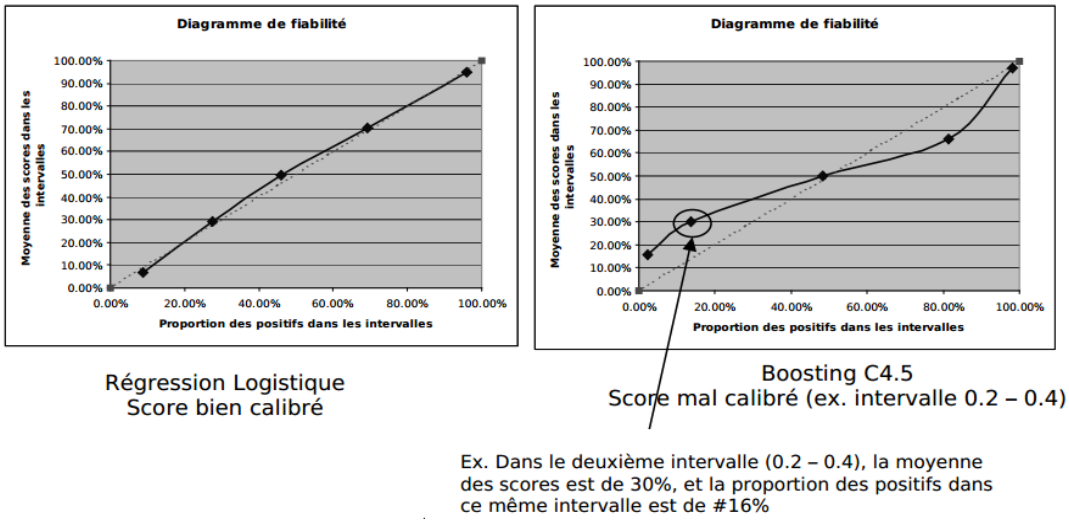


FIGURE 2.5 – Diagramme de fiabilité.

Il existe des méthodes de normalisation basées sur la mise en échelle des scores. Elles permettent de transformer une distribution de scores dans un intervalle donné, typiquement [0, 1]. Nous en citons :

– Min-Max : C'est une sorte de translation et de mise en échelle des valeurs.

$$s_i\prime = \frac{s_i - min}{max - min} \tag{2.22}$$

où min et max correspondent aux minimum et le maximum des scores. Les scores normalisés appartiendront à l'intervalle [0,1].

– Z-norm ou σ-norm : consiste à centrer les valeurs sur 0 et réduire la variance à 1. Les scores normalisés ne seront donc pas bornés.

$$s_i\prime = \frac{s_i - \mu}{\sigma} \tag{2.23}$$

où μ et σ correspondent à la moyenne et l'écart type des scores, et sont estimés sur un ensemble de développement.

– Tangente hyperbolique "tanh"

$$s_i\prime = \frac{1}{2}\{tanh(0.001 \times \frac{s_i - \mu}{\sigma}) + 1\} \tag{2.24}$$

où μ et σ correspondent à la moyenne et l'écart type des scores, et sont estimé sur un ensemble de développement. Les scores normalisés appartiennent à l'intervalle [0, 1].

Ces méthodes ont été utilisées et comparées dans l'état de l'art [SIYM03, JNR05]. Même si elles peuvent résoudre le problème des plages des valeurs, le problème de la disparité des distributions des valeurs demeure. Les comparaisons ne sont donc pas possibles. Certains travaux de recherche ont abordé ce problème et ont proposé des méthodes permettant de produire de bonnes valeurs de probabilité [NMC05, Pla99].

Au lieu de se contenter d'une simple remise à l'échelle, certaines autres méthodes de normalisation font des transformations plus sophistiquées, tout en considérant les classes des exemples. On peut en citer la méthode de Platt et la régression isotonique. La méthode de Platt [Pla99], aussi connue sous le nom de *la transformation sigmoïde*, consiste à transformer les scores avec une fonction sigmoïde :

$$s_i\prime = \frac{1}{1 + e^{a.s_i + b}} \tag{2.25}$$

où a et b sont des valeurs réelles. L'estimation des paramètres a et b se fait de manière à maximiser le log-vraisemblance V :

$$V = \sum_i y_i \times \ln(s_i) + (1 - y_i) \times \ln(1 - y_i) \tag{2.26}$$

où $y_i \in 0, 1$ est la classe (l'étiquette ou l'annotation) de l'exemple i. Les méthodes d'optimisations numériques (e.g. Newton) sont souvent sollicitées pour cet objectif.

Zadrozny et al. [ZE01] ont proposé une autre méthode, appelée "la régression isotonique", qui consiste en la réalisation d'une transformation via une fonction f^{iso} monotone croissante telle que :

$$y_i = f^{iso}(s_i) + \epsilon \tag{2.27}$$

avec :

$$y_i = \begin{cases} 1 \ si \ y_i = 1 \ (l'exemple \ est \ positif) \\ 0 \ si \ y_i = 0 \ (l'exemple \ est \ négatif) \end{cases} \tag{2.28}$$

Contrairement à la méthode de Platt qui optimise la vraisemblance, la régression isotonique optimise le critère des moindres carrés :

$$min \sum_i (y_i - f^{iso}(s_i))^2 \tag{2.29}$$

et donne la possibilité de choisir n'importe quelle fonction monotone croissante. En pratique, on utilise l'algorithme PAVA (Pool Adjacent Violators algorithm : [dLHM09]).

La transformation de Platt est plus efficace lorsque la distorsion dans les probabilités prédites a une forme sigmoïde. La régression isotonique quant à elle, est une méthode de calibration très puissante qui permet de corriger toute déformation monotone. Malheureusement, cette puissance supplémentaire a un prix. Une analyse de la courbe d'apprentissage montre que la régression isotonique est plus sensible au problème du sur-apprentissage, et reste ainsi moins efficace que la méthode de Platt lorsque les données sont rares [NMC05].

2.8 Fusion

Généralement un seul descripteur ne suffit pas pour avoir des performances satisfaisantes dans le cadre de la classification de données. En effet trouver un bon descripteur pour décrire chaque classe ou un concept reste un défi ouvert. Cette remarque reste vraie pour les données multimédia, pour lesquelles il y a plusieurs sources d'information. Par exemple, on peut décrire une image en se basant sur la couleur, la texture, ou en extrayant des points d'intérêts. Pour le cas des vidéos, plusieurs autres sources s'ajoutent à la liste : l'apsect temporel, les mouvements, l'audio, etc. Il est nécessaire de prendre en compte toute information utile pour une bonne description des données. Pour ce faire, des chercheurs ont étudié la combinaison des informations de diverses sources dans le but d'améliorer la performance de leurs systèmes d'indexation et/ou classification, appelée "fusion". Une définition générale de la fusion de données a été présentée par L. Wald [Wal98] : "La fusion de données constitue un cadre formel dans lequel s'expriment les données provenant de sources diverses ; elle vise l'obtention d'informations de plus grande qualité". Bloch et al. [BLCM03] quant à eux, proposent une autre définition : "La fusion d'information est la combinaison d'informations issues de multiples sources hétérogènes dans le but d'améliorer la prise de décision". La fusion peut être appliquée à deux niveaux différents :

1. Fusion de bas niveau, appelée "fusion précoce".
2. Fusion de haut-niveau, appelée "fusion tardive".

Il existe un autre méthode de fusion appelée "fusion de noyaux" qui concerne spécifiquement les approches d'apprentissage/classification à base de noyaux. Nous présentons dans la suite ces différents types de fusion.

2.8.1 Fusion précoce

La fusion précoce (*early fusion* en anglais) consiste à combiner un ou plusieurs descripteurs uni-modal(aux) pour générer une nouvelle représentation regroupant

des informations issues de différentes modalités. Dans ce cas, les informations fusionnées sont de brutes, c'est-à-dire proche du "signal". Dans le cadre de l'indexation et/ou classification, on parle de fusion de descripteurs de bas niveaux. Cela revient en pratique à concaténer plusieurs descripteurs de bas niveau extraits de différents médias (e.g. Audio, visuel) ou plusieurs descripteurs de différents types (e.g. Couleur, texture, etc) pour générer un nouveau vecteur de dimension plus grande. Ce nouveau descripteur est ensuite utilisé dans une approche d'apprentissage, comme présenté dans la figure 2.6(a). Nous notons à ce stade, qu'un seul apprenant est nécessaire pour l'apprentissage. Ce schéma de fusion est utilisé dans le cadre la détection de concepts dans les documents multimédia [Nap04, SWS05]. Sa simplicité et sa facilité de mise en œuvre fondent son succès et sa large utilisation par les communautés de la vison par ordinateur et du multimédia. Or, elle présente un inconvénient majeur qui est la taille du nouveau descripteur multi-modal. En effet cette dernière augmente en fonction du nombre de descripteurs uni-modaux pris en compte et aussi par leurs tailles, chose qui influence le processus d'apprentissage, menant souvent à des situations de non convergence de l'algorithme utilisé, ou à un long temps de calcul. En plus, certains descripteurs sont déjà de grandes dimensions, comme certains descripteurs basés sur les SIFTs qui peuvent avoir une taille d'environ 4000, ce qui pourrait faire exploser la taille du descripteur final. Comme alternative, il est nécessaire de recourir à des traitements de descripteurs et notamment les techniques de réduction de dimensionnalités (voir la section 2.6.2). D'autres problèmes auxquels se heurte la fusion précoce est la différence entre les plages de valeurs des différents vecteurs caractéristiques. Il est alors important d'appliquer une normalisation avant l'utilisation du descripteur résultant (voir la section 2.6.1). On peut rajouter comme problème la disparité entre les qualités d'informations fournies par les différentes modalités.

2.8.2 Fusion tardive

Contrairement à la fusion précoce qui combine des informations de bas niveau, la fusion tardive (*late fusion* en anglais) fusionne des informations sémantiques, qui sont souvent des scores de classification/prédiction ou de probabilités renvoyés par des apprenants entrainés sur différents descripteurs (ou n'importe quel autre type d'informations sémantiques). Comme illustré dans la figure 2.6(b), les scores de prédiction sont fusionnés en utilisant une simple fonction (moyenne, min, max, etc) ou via un nouvel apprenant (classificateur) dans une approche de "stacking" (voir la section 2.7.4.4. Nous soulignons ici que ce sont les résultats des prédictions qui sont fusionnés : ces derniers ne sont pas nécessairement liés à des différents descripteurs de bas niveau. En effet, on peut lancer plusieurs classificateurs différents en utilisant un même descripteur et fusionner les résultats obtenus : on est alors dans une approche d'apprentissage par ensemble (voir la section 2.7.4), comme dans [LJH02], où les auteurs utilisent un classificateur de type SVM et montrent que cela est plus efficace qu'une fusion à base du vote majoritaire. On peut utiliser également, le même classificateur avec différents descripteurs de même ou des modalités différentes [WCCS04], ou on peut fusionner

les résultats de différents classificateurs entrainés sur différents descripteurs. Ces trois cas résument un des avantages de la fusion tardive, qui est la possibilité de choisir un classificateur spécifique et adapté pour chaque modalité. D'autre part, la combinaison de prédictions de plusieurs apprenants induit à une décision plus précise, du fait que les différents classificateurs n'ont pas le même taux d'erreurs. L'inconvénient majeur de la fusion tardive est l'effort en apprentissage nécessaire. En effet, il y autant d'apprenants que de sources à fusionner, et donc, le même nombre de phases d'apprentissage. Dans [SDH+12], les auteurs proposent une fusion hiérarchique basée sur les étapes suivantes :

1. Fusion inter-classificateurs : fusion des scores obtenus par différents classificateurs ;
2. Fusion "par famille de descripteurs" : fusion des résultats de la première partie pour toute famille de descripteurs. Par exemple, en fusionnant des résultats pour les variantes de SIFT, VLAD, etc ;
3. Fusion mixte : fusion des résultats de l'étape précédente.

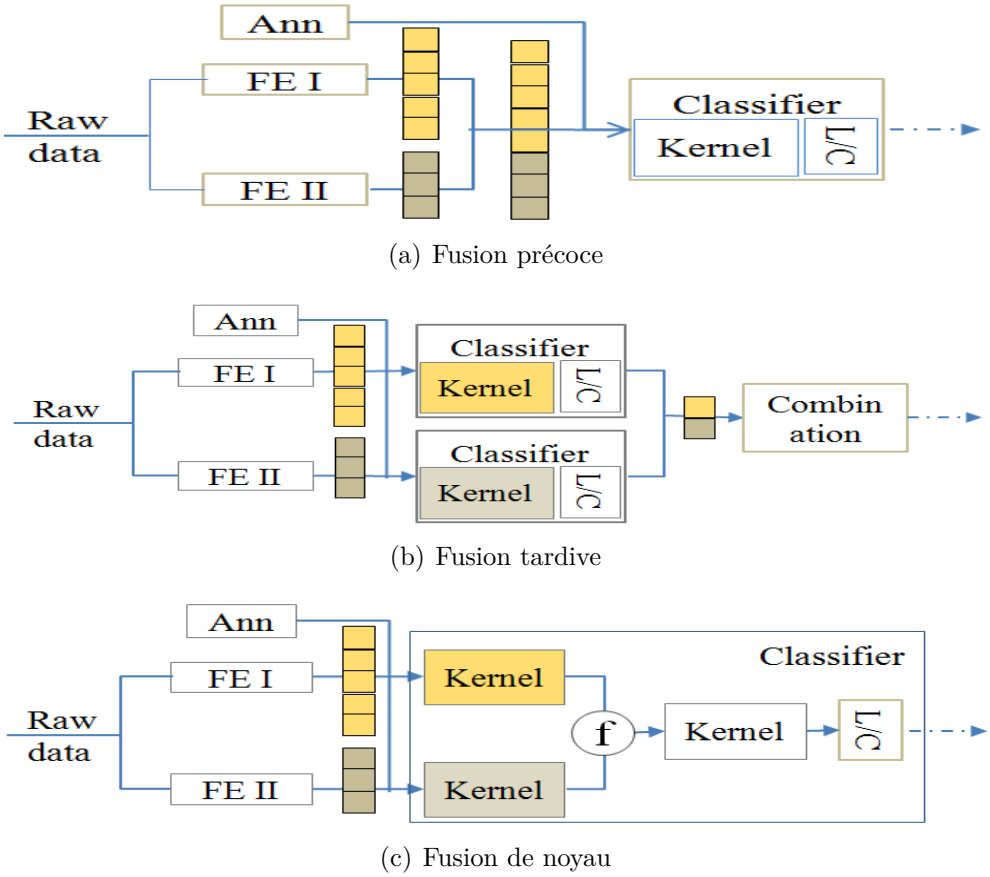
Cette fusion s'est montrée efficace dans la détection de concepts visuels dans les images et vidéos.

Snoek et al. [SWS05] ont fait une étude comparative entre la fusion précoce et la fusion tardive pour la détection de concepts visuels dans les vidéos. Ils conclurent que la fusion tardive est plus efficace que la fusion précoce, mais cela au détriment d'un effort important d'apprentissage. Cependant si la fusion précoce fait mieux que la fusion tardive le gain est plus significatif.

2.8.3 Fusion de noyaux

L'intérêt de l'utilisation d'un classificateur à base de noyau est de bénéficier des avantages de certaines propriétés utiles des noyaux. Cette méthode consiste en la combinaison de noyaux uni-modaux dans le but de générer un nouveau noyau multi-modal. Ce processus est illustré dans la figure 2.6(c). Cela permet de choisir un noyau uni-modal spécifique à chaque modalité. Par exemple, les histogrammes de couleurs peuvent bénéficier de l'avantage de certaines fonctions de distances spécifiques à la correspondance entre histogrammes. La modalité textuelle, quant à elle, peut être mieux traitée en utilisant des noyaux plus appropriés, comme les noyaux textuels [LSST+02]. La fusion de noyaux permet également de modéliser les données avec des paramètres plus adaptés. En effet, fusionner les modalités via une méthode de fusion précoce revient à modéliser les données en utilisant une seule fonction noyau. Par conséquent, en utilisant un noyau RBF, un seul paramètre σ est nécessaire pour apprendre les relations entre les différents échantillons. Cependant, il est plus judicieux d'utiliser un noyau RBF combiné en utilisant un paramètre σ par modalité [AQG07]. La combinaison de noyaux uni-modaux permet de garder le plus d'informations possible de chaque modalité. Un noyau RBF combiné prend la forme suivante :

$$K_c(x, y) = F(K_m(x_m, y_m)_{(1 \leq m \leq M)}) \tag{2.30}$$



(a) Fusion précoce

(b) Fusion tardive

(c) Fusion de noyau

FIGURE 2.6 – Différentes méthodes de fusion. Figure prise de [AQG07].

où $K_c(x, y)$ est la valeur du noyau combiné pour les échantillons x et y, $(K_m)_{(1 \leq m \leq M)}$ sont les noyaux RBF uni-modaux considérés, F est la fonction de fusion des M modalités. x_m et y_m sont les vecteurs représentant les échantillons dans la modalité m. Un des principaux enjeux de la recherche actuelle sur les noyaux est l'apprentissage de ce genre de noyaux combinés *Multiple Kernels Learning*. Le but est d'apprendre en même temps, les paramètres des noyaux uni-modaux et de ceux du noyau combiné [SRS06, GA11].

2.9 Ré-ordonnancement

L'indexation sémantique est généralement réalisée par un apprentissage supervisé, où le système est entrainé sur les échantillons positifs et négatifs par rapport à un concept cible (l'ensemble de développement) pour produire un modèle qui est alors utilisé pour la production de scores reflétant les probabilités que de nouveaux échantillons (l'ensemble de test) contiennent ce concept. Ces scores sont souvent calculés de façon homogène à une probabilité. La recherche peut alors être effectuée en classant les échantillons en fonction du score de probabilité, de façon à mettre dans le haut de la liste les échantillons les plus susceptibles de

contenir le concept cible. Il est souvent possible d'améliorer le rendement d'indexation ou d'extraction en modifiant le score de probabilité de l'ensemble des échantillons et ce, en utilisant les scores initiaux ainsi que d'autres sources d'information. Les nouveaux scores engendrent un nouveau classement des échantillons, d'où les noms de "reclassement" ou "re-ordonnancement (re-ranking en anglais)" ou "re-scoring". Beaucoup de travaux se sont focalisés sur ce genre d'approches, pour leur simplicité de mise en œuvre. Certains d'entre eux se basent sur l'utilisation d'information contextuelles, on verra en détails ce genre d'approches dans la section 2.11. D'autres approches utilisent un classificateur pour la génération de nouveaux scores de prédictions [KC07]. Des pseudo-étiquettes sont générées et utilisées en plus des résultats d'un système initial comme entrée du nouveau classificateur. Dans [YH08b], les auteurs proposent une méthode de reclassement ordinal qui reclasse une liste initiale de recherche en utilisant des modèles (patterns) de co-occurrences via des fonctions d'ordonnancement (e.g. ListNet). Certains travaux utilisent les relations entre les concepts qui sont basés sur des ontologies pour re-classer les résultats d'un premier traitement. [WTS04] utilisent les "ancêtres" des concepts dans une ontologie pour améliorer la détection initiale des concepts descendants. D'autres approches se servent de relations extraites depuis des ensembles de données [SNN03, ZWLX10, AOS07, NH01a, NH01b].

2.10 Les ontologies

Dans la philosophie grecque antique, une ontologie signifie la théorie de l'être, de l'existence et de la réalité. Le domaine de l'Intelligence Artificielle a pris ce terme pour désigner quelque chose d'un peu différent, mais par rapport à la signification originale du terme. Gruber [Gru93]définit une ontologie comme étant *une spécification explicite d'une conceptualisation d'un domaine de connaissance.*

Les ontologies ont été historiquement utilisées pour améliorer les performances dans les systèmes de recherche multimédia [WTS04, FGL07, WTS04]. Une ontologie est généralement formée d'un ensemble de concepts abstraits qui sont organisés en relations hiérarchiques entre eux. Il y a une différence sémantique dans le type de relation hiérarchique qu'un homme peut facilement distinguer, mais celle-ci (la relation hiérarchique) doit être explicitée pour la machine. Les concepts dans une ontologie sont d'une part liés à des mots lisibles par l'homme (littéraux) et d'autre part, ils sont liés entre eux par des relations sémantiques. Une ontologie peut être construite de différentes manières : 1) Manuellement, 2) Automatiquement, ou 3) De manière hybride. Plusieurs ontologies ont vu le jour et sont utilisées dans le domaine de la recherche d'information, on peut citer : Wordnet [2], Cyc [3] et LSCOM qui a été définie pour le domaine du multimédia. L'ontologie

2. WordNet est une ontologie lexicographique pour la langue anglaise [Fel98]. Elle est représentée sous la forme de listes liées entre elles pour créer un réseau. Elle est utilisée pour un dictionnaire (WordWeb2), un système expert (SearchAide), un logiciel d'annotation automatique des textes, etc. WordNet 1.7 a un réseau de 144 684 mots, organisés en 109 377 concepts appelées "synsets" [Ben09].

3. Le projet Cyc (qui dérive du mot "encyclopédie" lancé en 1984, cherche à développer une ontologie globale et une base de données de la connaissance générale, dans le but de per-

LSCOM-lite [NKK+05] est une version allégée de LSCOM [LSC06]. Elle a été développée dans l'atelier ARDA/NRRC[4] et contient les 39 concepts. LSCOM-lite a été utilisée par le NIST (National Institute of Standards and Technology) en collaboration avec les participants TRECVid. Plusieurs dizaines d'heures d'un total de centaines de milliers de plans vidéos ont été annotés par des concepts issus de LSCOM-lite.

L'ontologie a été exploitée pour le calcul de la similarité entre concepts en se basant sur sa structure. En effet, certains chercheurs ont proposé d'utiliser les arcs reliant les concepts pour mesurer la similarité sémantique entre ces derniers, et ce, en calculant la distance en termes du nombre d'arcs séparant les concepts [RMBB89]. Dans [RMBB89], la méthode *Edge Counting* estime la similarité entre deux concepts c_1 et c_2 par une mesure qui est proportionnelle à la distance minimale entre c_1 et c_2 dans la hiérarchie, comme le montre l"équation 2.31 :

$$sim(c_1, c_2) = \frac{1}{1 + dist_{rada}(c_1, c_2)} \tag{2.31}$$

où $dist_{rada}(c_1, c_2)$ désigne le plus court chemin entre c_1 et c_2 (i.e. Nombre minimum d'arcs séparant c_1 et c_2).

De manière similaire, d'autres travaux comme dans [FGL07], se sont basés dans le calcul sur le plus long chemin entre les concepts dans la hiérarchie.

D'autre part, certains chercheurs ont proposé des méthodes alternatives basées sur le contenu informatif des concepts pour le calcul de la similarité sémantique, en se basant sur une mesure entropique de la théorie de l'information [SC99, SVH04]. Nous retrouvons également dans l'état de l'art d'autres approches hybrides combinant les deux idées [Res99, Ben09].

2.11 Utilisation du contexte pour l'indexation sémantique

Cette thèse s'articule sur l'utilisation du contexte pour l'amélioration de la performance des systèmes d'indexation des images et vidéos. Plusieurs travaux de recherche ont tenté d'exploiter la notion du contexte dans plusieurs domaines, notamment celui de la vision par ordinateur. En effet, un atelier IEEE (Context-based vision CBVIS'95) concernant l'utilisation du contexte dans la vision, a eu lieu en 1995, où plusieurs travaux qui utilisent le contexte dans divers axes de recherche ont été publiés [BP95, BZ95, IB95]. La première problématique qu'on l'on rencontre pour instancier cette idée est de trouver une définition concrète et efficace de la notion du contexte. Une telle définition varie d'un travail à un autre selon le domaine concerné et le type de données manipulé. Il s'ajoute à

mettre à des applications d'intelligence artificielle de raisonner d'une manière similaire à l'être humain [MWK+05, Ben09].

4. Northeast Regional Research Center.

cela le problème de trouver une méthode efficace pour l'exploitation des données contextuelles.

Dans cette section, nous allons présenter la notion du "contexte", les différents types de contextes pouvant être utilisés ainsi que les différentes manières possibles d'utilisation du contexte dans le cadre de l'indexation sémantique des images et vidéos.

2.11.1 Définition du contexte

Il est difficile de donner une définition précise du contexte, puisque cette dernière varie selon le domaine abordé. Avant de choisir et de présenter notre définition retenue dans le cadre de ce travail, nous allons exposer dans ce qui suit certaines définitions utilisées par des chercheurs en informatique et d'autres.

Wikipédia :

« *Le contexte d'un évènement inclut les circonstances et conditions qui l'entourent ; le contexte d'un mot, d'une phrase ou d'un texte inclut les mots qui l'entourent.* »

Larousse :

« Le contexte est un :

- *ensemble du texte à l'intérieur duquel se situe un élément d'un énoncé et dont il tire sa signification* ;
- *ensemble des éléments (phonème, morphème, phrase, etc.) qui précèdent et/ou suivent une unité linguistique à l'intérieur d'un énoncé* ;
- ... ; »

Bien que le terme "contexte" est fréquemment utilisé dans la vision par ordinateur, on signale le manque d'une définition claire [DHH+09]. Le contexte est vaguement vu comme « *n'importe qu'elle et toute information susceptible d'influencer la manière dont une scène et les objets qui s'y trouvent sont perçus.* » [Str93, DHH+09].

Brézillon [Bré99] énumère un ensemble de définitions différentes du contexte et les regroupe selon plusieurs domaines. Nous avons choisi certaines concernant le domaine de la vision.

Selon Brézillon, le contexte est un facteur important dans des applications différentes dans le domaine de la vision : la reconnaissance de caractères [Tou78], la reconnaissance d'objets [AVFRP84], etc.

Dey [Dey01] propose la définition suivante : « *Le contexte est toute information qui peut être utilisée pour caractériser la situation d'une entité. Une entité est une personne, un lieu ou un objet qui est considéré(e) comme pertinent(e) pour l'interaction entre un utilisateur et une application, y compris l'utilisateur et les applications eux-mêmes.* ».

Winograd [Win01] dit que : « *quelque chose est un contexte par rapport à la façon dont il est utilisé dans l'interprétation.* ».

Dans [GB10], Galleguillos et Belongie considèrent dans le domaine de la catégorisation d'objets la définition suivante : « *La connaissance contextuelle peut*

être toute information qui n'est pas directement produite par l'apparition d'un objet. Elle peut être obtenue à partir des données entourant l'image, des étiquettes ou des annotations de l'image et la présence et l'emplacement d'autres objets. ».

Brémond et Thonnat [BT97] proposent une définition du contexte à travers la description des différents types d'informations manipulés par un processus. Les auteurs illustrent l'intérêt de leur définition avec un exemple de processus d'interprétation d'une scène. Un principal résultat de leur travail est d'avoir une représentation donnée sous différents points de vue [Bré99].

Dans le cadre des applications conscientes du contenu ou du contexte, Schilit et al. [SAW94] déclarent que : « *les aspects importants du contexte sont : où vous êtes, avec qui vous êtes, et quelles sont les ressources à proximité.* ».

Desvignes et al. [DPS91] proposent le système "SISI" permettant d'exploiter le contexte et son rôle dans l'interprétation d'une séquence d'images. Les auteurs définissent le contexte comme étant :« *l'ensemble de propriétés qui sont associées à une entité en fonction de l'environnement dans lequel cette entité se trouve.* ».

Aucune définition d'entre celles évoquées précédemment ne coïncide avec notre problématique, qui rappelons le, s'articule sur la détection de concepts visuels dans les images et vidéos. Or, bien qu'elles soient différentes, la majorité de ces définitions se rejoignent pour partager quelques points communs. En l'occurrence, les points suivants :

1. le contexte est un ensemble d'informations *supplémentaires* par rapport à un objet, un processus (e.g. Détection d'objets dans les images) ou une donnée (e.g. Un mot dans un texte) ;
2. le contexte se trouve dans un voisinage donné : autour d'un mot dans du texte, autour d'un objet dans une image, ou déduit à partir d'une situation ;

Nous ne focalisons pas dans notre travail sur les concepts de type objets, mais nous considérons n'importe quel type de concepts : objets, évènements, etc. Ce choix nous impose de considérer une définition générale et non spécifique. Nous choisissons donc, de retenir la définition suivante qui repose sur les deux points en communs évoqués ci-avant : « *le contexte est toute information additionnelle qu'un système d'indexation de base peut s'en passer, qui est pertinente et peut aider à améliorer la qualité de l'indexation.* ». Ceci dit, si la présence ou l'absence d'une information est pertinente pour un système par rapport à son objectif final, alors cette information est considérée comme un contexte.

2.11.2 Pourquoi avoir besoin du contexte ?

Le contexte aide à comprendre le sens d'un mot et permet de désambiguïser des termes polysémiques. Beaucoup de recherches ont profité de l'avantage de cette notion dans la recherche d'information. Globalement, il est reconnu que l'utilisation du contexte permet de concevoir des algorithmes d'analyse et de compréhension d'images moins complexes et plus robustes [Bré99]. Pour l'indexation et la recherche des images/vidéos par détection de concepts, cette idée semble a priori valable. D'autre part, il n'est pas toujours évident d'avoir des

apprenants efficaces quelque soit la classe ou le concept à traiter. En effet, certains concepts s'avèrent difficiles à modéliser à cause de l'inexistence de bons descripteurs permettant de les représenter efficacement ou à cause du manque de données annotées, surtout pour la classe positive. Par conséquent, utiliser des informations contextuelles peut être très utile pour corriger des erreurs de classifications ou de renforcer les décisions des apprenants ou changer l'ordre des résultats en termes de pertinence par rapport au concept recherché. La figure 2.7 illustre un exemple montrant l'utilité du contexte pour la désambiguïsation, la correction des erreurs et la détection de nouveaux concepts sémantiques dans les images et vidéos. L'étape 1, où il s'avère difficile de savoir ce que contient l'image, peut représenter le cas d'une mauvaise détection de concepts. Savoir que cette image contient les concepts "Route" et "Dehors" aide à déduire que l'image contient probablement un "véhicule" et que la partie blanche dans l'image correspond à de la "neige". De même, savoir que l'image de notre exemple contient les concepts "Montagne" et "Personne", aide à rejeter l'existence d'un véhicule et renforcer l'hypothèse qu'il s'agit de l'occurrence d'un "Skieur" et probablement l'apparition du concept "Ciel". Dans les deux cas, la probabilité d'apparition du concept "Réfrigérateur" est diminuée, voire annulée.

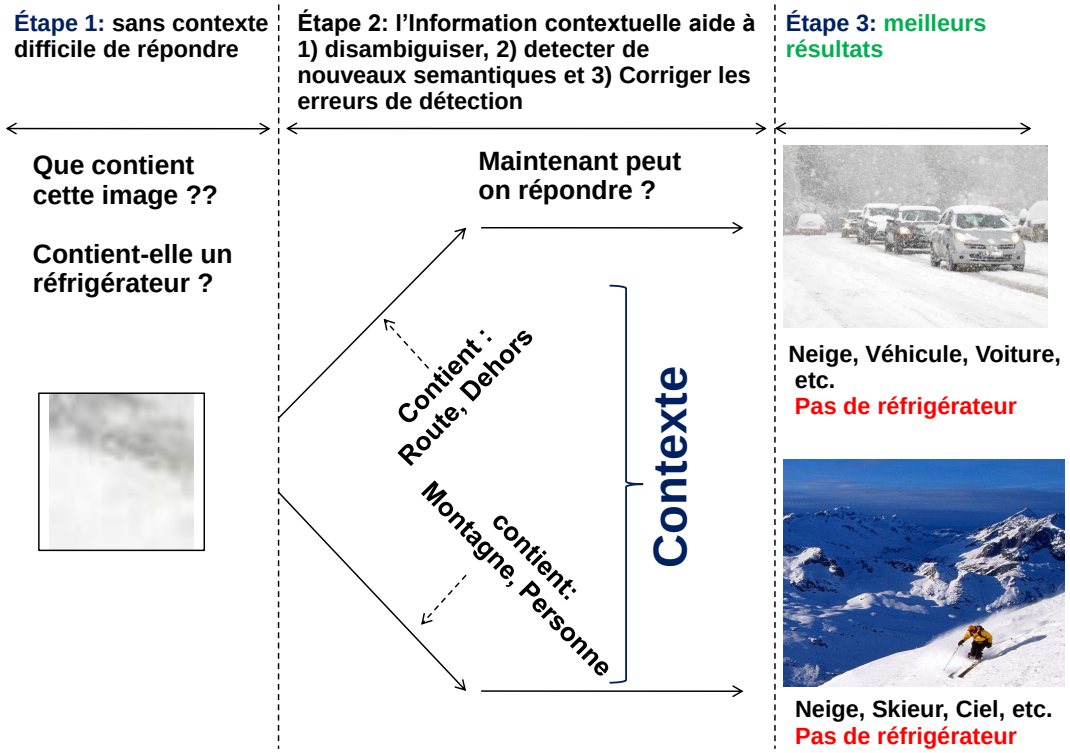


FIGURE 2.7 – Utilité du contexte pour la détection des concepts visuels.

D'autre part, les concepts n'existent pas indépendamment les uns des autres dans les images et vidéos. En effet, même les concepts qui n'ont rien à voir avec le concept cible peuvent être utiles pour améliorer la performance des détecteurs. Par exemple, l'apparition du concept "Cuisine" va aider à rejeter l'occurrence de

plusieurs concepts tels que “Camion”, “Terrain de foot”, etc. Tenter de détecter des concepts sans tenir compte des relations les reliant à d’autres concepts est une idée naïve.

En outre, la présence de certains concepts peut avoir des caractéristiques spécifiques. Par exemple, le concept “ciel” apparaitra toujours en haut d’une image, la “mer” en revanche sera toujours en bas. L’événement de “course” ou de “poursuite de police”, va s’étendre sur plusieurs plans vidéo. Ces informations peuvent être utilisées comme source de contexte pour améliorer la performance d’un système de détection de concepts.

[GB10] déclarent que des études récentes en psychologie et en cognition montrent que le contexte sémantique facilite la reconnaissance visuelle dans la perception humaine.

Dans son article [Abb10], Sunitha affirme qu’un objet vidéo séparé de son contexte n’est pas complet et qu’un système de recherche d’information n’est complet que s’il renforce les concepts sémantiques avec les détails du contexte. En effet, les utilisateurs veulent rechercher une information selon son contenu ou son contexte sémantique, ils cherchent à avoir le résultat en décrivant juste ce qu’ils veulent et idéalement en exprimant cela en langage naturel. Pour ce faire, le système doit disposer de connaissances concernant la sémantique et le contexte. Ces connaissances complètes peuvent être exprimées par une ontologie.

Selon Desvignes et al. [DPS89], le rôle du contexte dans un système de vision est : guider la recherche, la résolution des ambiguïtés, combler les lacunes, corriger les erreurs et l’apprentissage.

Bien qu’il soit intuitivement correcte que les relations contextuelles peuvent aider à améliorer la précision des détecteurs individuels de concepts visuels, des expérimentations ont montré que cette amélioration n’est pas toujours stable et que la performance globale peut même être moins bonnes que celles des détecteurs individuels (sans utilisation du contexte) [QHR+07]. Par exemple, dans [HyCC+04], au moins trois des détecteurs des huit concepts considérés n’ont pas gagné en performance en utilisant la fusion conceptuelle avec un classificateur de type régression logistique.

2.11.3 Catégorisation du contexte

Nous pouvons distinguer différentes catégories de contexte dans le domaine d’indexation des documents multimédia, en fonction de l’origine de l’information contextuelle. De ce fait, on peut catégoriser le contexte selon trois catégories :

- Le contexte environnemental : concerne toute information qui se trouve dans les alentours d’une image ou une vidéo. Par exemple, sur le site youtube [5], cela peut concerner tout le texte qui entoure la vidéo, la description des utilisateurs de la vidéo, les commentaires, etc ;
- Le contexte de création et d’utilisation : concerne toute information qui a un lien avec la création du document. Par exemple, les méta-données, le

5. YouTube est un site web d’hébergement de vidéos sur lequel les utilisateurs peuvent envoyer, regarder et partager des séquences vidéo (Wikipédia). Lien du site : http://www.youtube.com/

Contextes	Sources
Contexte local basé sur les pixels	La fenêtre d'entourage/encadrement, voisinages dans une image, frontière/forme d'un objet. [WB06].
Contexte de fond d'un scène 2D	Les Statistiques globales d'une image. [OT01, RTL+07]
Contexte géométrique 3D	Le plan 3D d'une scène, la surface d'appui [HEH07, DAM08], les orientations de surface, des occlusions [HEH11], des points de contact, etc
Contexte sémantique	Les évènements, les catégories d'une scène, les objets présents dans une scène et leurs étendus spatiaux, les mots clés
Contexte photogrammétrique	La hauteur et l'orientation de la caméra [DAM08], la longueur focale, la distorsion de l'objectif, la fonction de réponse radiométrique
Contexte d'illumination	La direction du soleil [lal], la couleur du ciel, la couverture nuageuse, le contraste de l'ombre, etc
Contexte de Météo	La vitesse du vent / direction, la température, la saison, etc. [lal, LNE10, NN02]
Contexte géographique [HE08]	La localisation GPS, le type de terrain, la catégorie d'utilisation des terres, l'altitude, la densité de population, etc
Contexte temporel	Les plans ou les images voisin(e)s (si vidéo), les images temporellement proximales, les vidéos de scènes similaires [LYT+08], le temps d'acquisition [GNC+08]
Contexte culturel [GC08]	Le biais du photographe, le biais de sélection de données [PBE+06], les clichés visuels, etc

Table 2.2 – Quelques sources d'informations contextuelles [DHH+09].

propriétaire de l'image/vidéo, la date de la création du document, l'identité de la personne qui a téléchargé la vidéo. L'identité des personnes qui ont visionné l'image/vidéo, le nom du fichier assigné au document, la source où le document a été piqué, etc ;

– Le contexte interne ou du contenu : toute information ayant un lien avec le contenu de l'image ou de la vidéo. Par exemple, pour l'image, on peut considérer les différentes parties spatiales comme étant un contexte. Pour la vidéo, en analysant son contenu visuel, l'audio ou la transcription de la parole, peuvent être vus comme une source de contexte. Ces informations sont en général de bas niveau. Or, elles peuvent également être de haut niveau (sémantique), comme par exemple, les informations sur la catégorie du contenu : photos de personnes, reportage, sport, etc ;

D'autre part, chacune de ces catégories peut être structurée en sous-catégories. Par exemple, le contexte interne peut être classé selon la modalité qu'il considère ou de laquelle il est extrait : audio, visuel, textuel, sémantique, ou également selon le type d'information : spatiale, sémantique, temporelle, ou aussi selon la nature des images/vidéos : vidéos de sport, vidéo de cuisine, photo de mariage, etc. Le tableau 2.2 résume quelques types de contexte considérés dans le domaine de la vision par ordinateur.

Dans l'état de l'art, le contexte a été catégorisé en plusieurs classes. En effet, selon le niveau depuis lequel il est extrait, on peut parler de contextes global et local [GB10]. Le contexte global est généralement basé sur des statistiques concernant l'image entière ou un plan vidéo entier. Le contexte local quant à lui, est extrait localement depuis une partie ou une région d'une image/plan vidéo ou de l'entourage des objets contenus dans une scène [Tor03]. Le contexte local peut capturer différentes relations locales comme les interactions entre les pixels, les régions et les objets [CdFB04, SWRC09].

Le contexte global est préférable sur le contexte local pour son efficacité et sa simplicité d'extraction, puisqu'il ne nécessite pas de traitement lourds, comme la division de l'image en sous-régions et le traitement de chaque région localement. Cependant, il s'avère inefficace quand la scène contient beaucoup d'objets. D'autre part, le contexte global est parfois basé sur des statistiques calculés sur l'arrière plan, ce qui rend cette idée inefficace dans la situation où de grands objets apparaissent et couvrent de grandes parties de la scène, ne laissant que de petits espaces rétrécis dans lesquels apparait l'arrière plan.

Dans le domaine de la vision par ordinateur et de l'indexation multimédia, plusieurs types de contextes sont utilisés. Parmi les plus célèbres, on peut citer les contextes sémantique, temporel, spatial, le contexte d'échelle, etc. Chacun de ces types est plus important que les autres en fonction de la tâche considérée. En effet, le contexte spatial s'avère crucial pour l'annotation locale des images, comme décrit dans [YLZ07].

Pour reconnaître les catégories de scènes naturelles, [LSP06] ont proposé d'extraire "une pyramide spatiale", qui consiste à diviser l'image en sous-régions de plus en plus fines et de calculer des histogrammes de caractéristiques locales extraites de chaque sous-région.

[QHR+10] proposent de modéliser le contexte spatial entre des régions d'une même image via un modèle de Markov caché bi-dimensionnel dans le but de discriminer des classes.

[MEA10] abordent la problématique de recherche de vidéos en l'absence d'annotations. Les auteurs utilisent des relations spatiales qualitatives du genre : "à droite de", "à gauche de", "en dessous de", "en dessus de", "tomber dans le même carré d'une grille".

[Bar04] examinent la conséquence de l'utilisation des relations spatiales entre des objets qui co-occurrent dans la même scène. Ils ont conclu que : 1) La présence d'objets qui ont une interprétation unique améliore la reconnaissance d'objets ambigus dans une scène, et 2) Les relations spatiales entre les objets permettent de diminuer le taux d'erreurs dans la reconnaissance d'objets individuels.

[FE73] proposent de segmenter l'image en régions. Le contexte spatial est défini comme étant l'emplacement relatif des objets. Ensuite, ils vérifient la plage de relations spatiales qui doivent être satisfaites pour que l'objet soit présent.

Dans [SWRC09], les auteurs proposent d'utiliser un classificateur pour capturer les interactions spatiales entre les classes de pixels dans un voisinage, pour les incorporer ensuite dans un modèle de type "Conditional Random Rield (CRF)".

Un autre type de contexte a été pris en compte dans le domaine de la vision par ordinateur pour la catégorisation d'objets, appelé "contexte d'échelle" (Scale context en anglais). Le contexte d'échelle se base sur les relations d'échelles ou de tailles entre un objet et certains autres. Ceci dit, ce type de contexte exige qu'au moins un autre objet soit détecté et qu'un traitement soit réalisé de façon à déterminer les relations spatiales, de tailles et/ou de profondeurs entre l'objet cible et les autres objets l'entourant qui ont été détectés [GB10]. Strat et Fischler [SF91] intègrent le contexte d'échelle dans leur système "CONDOR" de reconnaissance d'objets. L'information contextuelle pour un objet donné consiste en un ensemble de méta-données d'une caméra regroupant la position et l'orientation de la caméra et certaines autres informations géométriques et de positionnement.

Dans [Tor03], l'auteur considère comme contexte d'échelle, des caractéristiques apprises à partir d'un ensemble d'images d'apprentissage en se basant sur la corrélation entre des caractéristiques de bas niveaux et la scène entière.

Lorsque les unités indexées sont des plans/images, les chercheurs se sont concentrés sur l'exploitation de deux types importants de contexte, comme le montre la figure 2.8 : 1) le contexte temporel entre les plans d'un même document vidéo, et 2) le contexte conceptuel ou sémantique entre les différents concepts cibles (dimension sémantique). Contrairement à la dimension temporelle, la dimension conceptuelle n'est pas réellement linéaire, parce que les concepts ne peuvent pas être naturellement ordonnés et organisés de façon linéaire. Ces deux types de contexte constituent cependant des dimensions distinctes et orthogonales (tous les concepts cibles doivent être indexés pour tous les plans). Les deux dimensions peuvent être considérées et utilisées séparément ou conjointement pour améliorer les performances d'un système. Cependant, pour l'indexation des images fixes, si on omet la disponibilité de toute donnée supplémentaire, seule la dimension sémantique peut être utilisée.

Généralement, le contexte sémantique est pris en compte par défaut par le contexte spatial et d'échelle, à partir du moment où les informations relatives à ces deux derniers sont calculées par rapport à l'occurrence d'autres objets/concepts. Donc, la co-occurrence des objets/concepts encodent déjà de l'information sémantique entre ces objets ou concepts.

Parmi les approches de l'état de l'art qui considèrent la dimension conceptuelle, on peut mentionner le travail de Naphade et al. [RNKH02], dans lequel les concepts sont organisés via un graphe et les relations entre eux sont modélisées avec une approche bayésienne (l'approche *Multinet*). Bien que plusieurs méthodes consistent principalement en un post-traitement de scores de classification qui sont obtenus de classificateurs indépendants, Qi et al. [QHR+07] ont proposé de

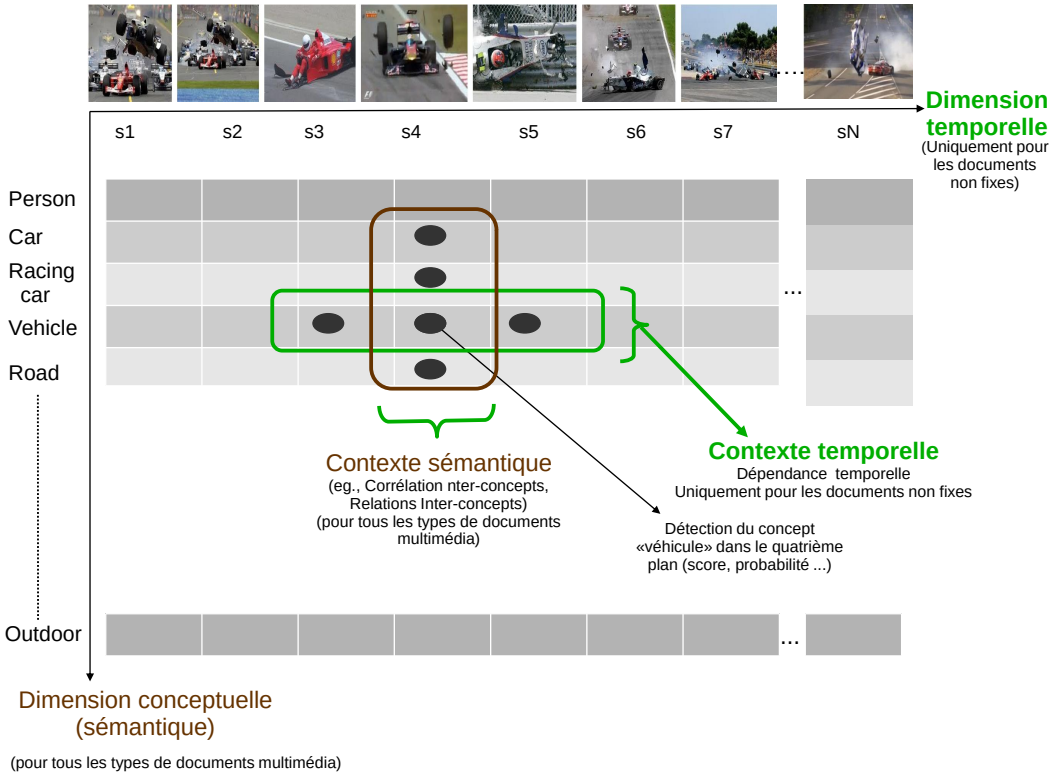


FIGURE 2.8 – Les deux dimensions de contexte les plus considérées dans l'indexation sémantique des documents multimédia.

générer un classificateur et de l'entrainer directement sur les scores de détection de tous les concepts cibles.

Dans [WB06], les pixels de chaque image sont annotés de manière à préciser si le pixel appartient ou pas à un objet contenu dans l'image. Le contexte est ensuite présenté sous forme de listes d'annotations par pixel indiquant l'occurrence du pixel dans les objets.

Le contexte est généralement extrait à partir de données d'apprentissage fortement annotées. Or, il peut être aussi obtenu de sources de connaissances externes comme c'est le cas dans [RVG+07], où les auteurs interrogent l'application web de Google : "Google Sets web application". Cette application génère une liste d'items potentiellement liés à partir de quelques exemples, en fournissant une matrice de co-occurrence binaire, où chaque entrée indique si un objet co-occurre ou pas avec un autre objet. L'application "Google Set" n'est malheureusement plus disponible depuis Septembre 2011.

Pour annoter des clips vidéo, [QHR+07] proposent d'apprendre des modèles de détection de concepts tout en modélisant et en tenant en compte les relations entre les concepts dans un seul et même stage, dans une approche corrélative multi-étiquettes (multi-label). Les auteurs concatènent le descripteur de bas niveaux avec un descripteur tenant en compte les corrélations inter-concepts et obtiennent de bonnes performances. Dans [QHR+08], l'approche a été étendue

pour incorporer des informations temporelles issues de modèles de Markov cachés entrainés sur des séquences vidéos.

[QGWF10] considère le contexte sémantique extrait depuis les sous-titres des vidéos pour raffiner les annotations d'images. La probabilité d'annotation d'un plan par un concept c est augmentée si ce concept est sémantiquement similaire ou corrélé au contexte extrait depuis le plan concerné. Les auteurs ont évalué leur approche sur la collection TRECVid'05, et ont utilisé comme sous-titres les résultats d'un système de reconnaissance de la parole, fournis avec la même collection. Leur proposition a donné une amélioration de la performance, mais ils ne l'ont pas comparée à des méthodes de l'état de l'art.

[GB10] abordent le problème de l'intégration du contexte pour la catégorisation des objets et font une revue de différentes manières d'utiliser l'information contextuelle pour la catégorisation des objets en prenant en compte les niveaux les plus courants d'extraction du contexte et des interactions contextuelles.

Les vidéos ont une caractéristique qui les différencie des images fixes ou d'autres documents : l'aspect temporel. Ignorer cette caractéristique entraîne une perte d'informations pertinentes. En effet, pour comprendre le contenu d'une vidéo, l'humain a besoin des informations contenues dans un intervalle de plans successifs, car les plans successifs d'une vidéo sont sémantiquement et temporellement liés. Sur la base de cette idée, nous supposons que la présence d'un concept dans un plan augmentera la probabilité de sa présence dans certains plans le suivant ou le précédant, en particulier pour les concepts qui apparaissent pendant une longue période, comme par exemple, les événements qui durent généralement longtemps.

Plusieurs chercheurs ont tenté d'utiliser la dimension temporelle pour l'indexation et/ou la recherche sémantique des vidéos. Cependant, dans l'état de l'art, trois possibilités différentes d'utilisation de l'information temporelle sont considérées, comme le montre la figure 2.9. Nous les décrirons dans les paragraphes suivants.

La façon la plus simple d'utiliser la dimension temporelle est de considérer une étape de ré-annotation (re-scoring ou re-classement) par post-traitement des résultats de l'étape de classification (cf. Figures 2.9 et 2.10). Dans ce contexte, [WM09, SQ11c, SQ11b, HQM12] proposent de mettre à jour le score initial sc_{ij} de détection d'un concept cible dans le plan j d'une vidéo v_i en prenant en compte des informations sémantiques déduites depuis la vidéo entière (cf. Figure 2.10). Dans [SQ11c], les auteurs proposent de calculer un nouveau score de détection sc'_{ij} en utilisant la formule suivante :

$$sc'_{ij} = sc_{ij}^{1-\gamma}.\ z_{ij}^{\gamma}, \tag{2.32}$$

où γ est un paramètre qui commande la robustesse de la méthode de reclassement ; il est réglé par validation croisée sur la collection de développement, et z_{ij} est un score global calculé à partir des scores des plans voisins via une fonction simple telle que la moyenne arithmétique, la moyenne géométrique, imin, max, etc.

Cependant, la taille w de la fenêtre est un paramètre qui doit être optimisé. Pour ce faire, on peut choisir entre une "ré-annotation globale", en fusionnant pour

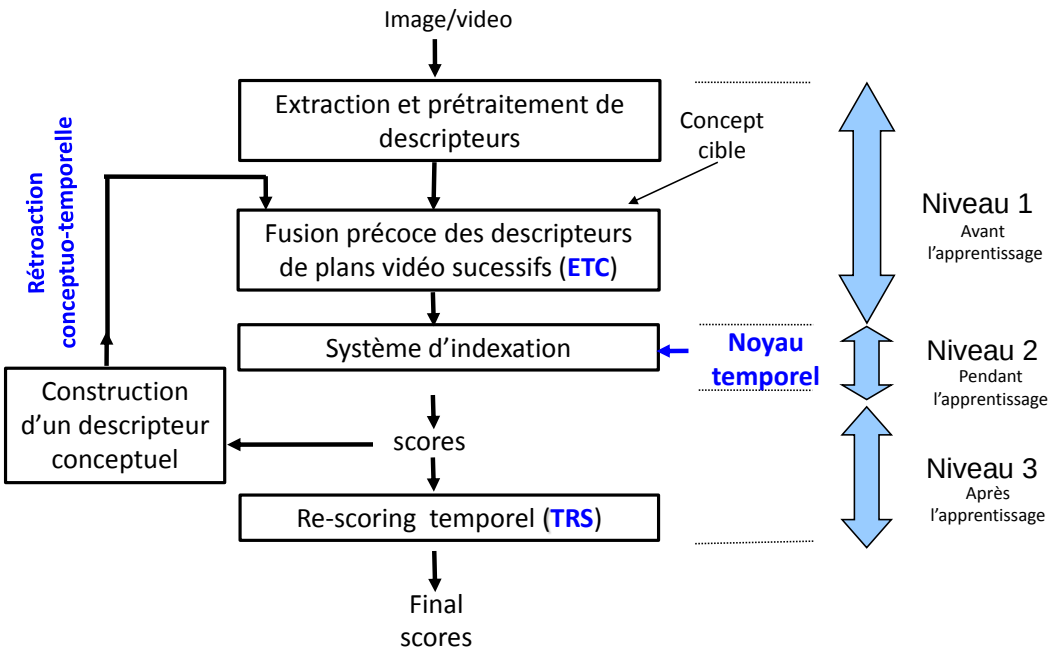


FIGURE 2.9 – Différentes utilisations possibles du contexte temporel dans l'indexation sémantique des vidéos.

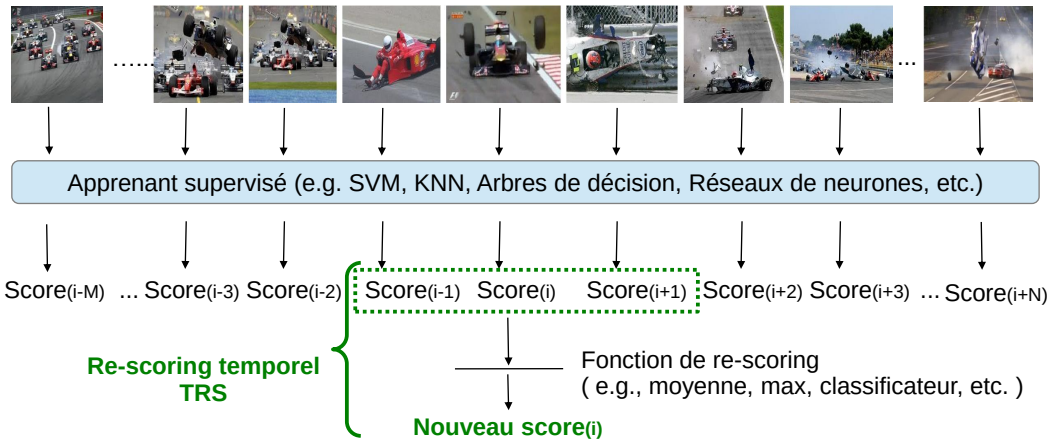


FIGURE 2.10 – Re-scoring temporel pour l'indexation sémantique des vidéos.

chaque plan les scores de tous les plans appartenant à la vidéo en question (i.g. la taille de la fenêtre = taille globale de la vidéo), et une "ré-annotation locale" en considérant la taille d'une fenêtre plus petite que la taille de la vidéo traitée. Ce choix dépend de la nature des vidéos. Il a été montré que pour des vidéos longues et hétérogènes, la "ré-annotation locale" est mieux adaptée, tandis que la "ré-annotation globale" est préférée pour les vidéos courtes et homogènes [SQ11c, SQ11b].

Le contexte temporel peut être également considéré en utilisant des noyaux temporels [QHR+08]. L'utilisation de l'information contextuelle est incorporée dans l'algorithme d'apprentissage. Le problème avec ce genre de méthodes, c'est

qu'elles ne sont pas faciles à mettre en œuvre, et nécessitent généralement un long temps de calcul. Dans [QHR+08], une fonction noyau temporelle a été proposée :

$$K(x, \tilde{x}) = \exp\{-\frac{d(x, \tilde{x})}{\sigma^2}\} \tag{2.33}$$

où $d(x, \tilde{x})$ est une fonction de distance entre deux vecteurs caractéristiques de bas niveaux x et $\tilde{x}$, et σ est le rayon du noyau

Les auteurs suggèrent d'utiliser la divergence de Kullback-Leibler (*Kullback-Leibler Divergence (KLD)*), parce qu'elle est une mesure de distance bien définie dans la théorie d'information [CT91]. Pour capturer les dynamiques temporelles des séquences vidéo, ils ont choisi de construire des modèles de Markov cachés (HMMs) afin de calculer la KLD entre ces modèles dynamiques.

Dans [SSPS09], les auteurs proposent une technique d'annotation basée sur la redondance du contexte. Leur idée part du principe que plus une vidéo v_i étiquetée t_i inclut une séquence plus longue d'une vidéo v_j étiquetée t_j, plus la probabilité que v_i soit étiquetée t_j augmente. Autrement dit, le but est une auto-annotation de vidéos basée sur la ressemblance inter-vidéos. Leur méthode cherche le chevauchement entre deux vidéos. Pour calculer le taux de chevauchement entre les vidéos, les auteurs ont construit un système de détection de copies de vidéos basé sur le modèle présenté dans [PD07a].

Pour l'annotation de vidéos, [LTD+10] proposent de considérer les contextes spatial et temporel en employant un noyau prenant en compte la corrélation temporelle (la consistance temporelle d'un concept et la dépendance temporelle entre des concepts distincts) et la corrélation spatiale (interaction entre les concepts dans un même plan vidéo). Par exemple, quand le concept c_1 est présent dans trois plans successifs, les deux concepts c_2, c_3, co-occurrent dans les deux derniers plans vidéo successifs. Des expérimentations sur la collection TRECVid'05 et TRECVid'07 ont montré des résultats intéressants.

Les deux dimensions conceptuelle et temporelle peuvent être considérées simultanément. Par exemple, Qi et al. étendent leur approche pour prendre en compte la dimension temporelle via l'utilisation d'un noyau temporel [QHR+08]. Weng et al. [WC12] ont également proposé une méthode directe qui prend en compte les deux dimensions en se focalisant sur le problème d'adaptation inter-domaines. Les méthodes qui considèrent les deux dimensions à la fois sont théoriquement capables de capturer toutes les relations "conceptuo-temporelles" disponibles (e.g. *Le concept A apparait n plans après le concept B avec une probabilité p*) entre les concepts dans un modèle statistique unique. Cependant, cette catégorie d'approches est plus exposée au problème du sur-apprentissage et requière une grande quantité de données d'apprentissage pour entrainer efficacement le modèle.

Dans [WC], les auteurs proposent d'intégrer le contexte sémantique (corrélation inter-concepts) et le contexte temporel (la dépendance entre les plans de la vidéo) pour la détection de concept visuels dans les vidéos. Pour créer les relations contextuelles entre les concepts, un algorithme similaire aux arbres de décision à été introduit. Leur approche se base sur la corrélation inter-concepts calculée via le test "chi-square" sur des ensembles de données mis à jour au fur

et à mesure en subdivisant l'ensemble de données initial. Des relations temporelles (inter-plans de vidéos) sont détectées de la même manière que les relations inter-concepts. Un parcours chronologique dans les deux sens (vers l'avant, vers l'arrière) est effectué ; l'opération continue jusqu'à ce qu'aucun plan ne présente une corrélation significative. Une fusion de trois informations est réalisée pour calculer la probabilité de l'occurrence d'un concept dans un plan vidéo s_t : la probabilité de l'occurrence du concept dans le plan s_t, la corrélation inter-concept et la corrélation temporelle.

2.11.3.1 Choix des types de contexte retenus dans le cadre de cette thèse

Nous rappelons qu'un de nos objectifs de départ était d'opter pour des approches contextuelles génériques pouvant être appliquées à n'importe quel système d'indexation. Nous avons donc pris ce critère en considération pour les choix des méthodes. Notre argument dans ce contexte est le fait que faire des choix très spécifiques réduirait le cercle d'applicabilité de nos contributions. Nous avons choisi de ne pas considérer un type spécifique de concepts (e.g. Objets, visages, etc). Nous avons vu dans la section 2.11.3 qu'il y a plusieurs catégories de contexte pouvant être exploitées dans un système d'indexation d'images et/ou de vidéos. Il s'est avéré très difficile d'une part, de tester tous les types de contexte, et d'autre part, la sélection de certaines catégories n'était également pas une tâche facile. En effet, les différents types de contexte ont montré leur utilité dans l'état de l'art, en atteignant de bonnes performances. Cependant, nous pouvons constater que certains contextes ne sont pas exploitables quelque soit la problématique traitée. Par exemple, le contexte local, et plus spécifiquement les contextes spatial et d'échelle, sont plus adaptés à la détection de concepts de type objets. Par conséquent, nous avons pensé que certains types de concepts abstraits, surtout ceux qui concernent les évènements, ne peuvent pas bénéficier de ces types de contexte. D'autre part, certains autres types de contexte présentent la difficulté d'obtention des informations contextuelles, comme par exemple, les métadonnées, le contexte géographique, le contexte relatif à l'appareil d'acquisition, etc. En effet, on n'a pas nécessairement accès à ce genre d'informations pour n'importe quel document multimédia. Ces raisons, en plus du fait que nous visons des approches génériques, nous ont rétréci le champs de sélection. En faisant un point sur ce qui a été fait dans l'état de l'art, et en tenant en compte le temps dont on disposait ainsi que la lourdeur du protocole expérimental à mettre en œuvre, nous avons conclu qu'il y a deux pistes à explorer, sur lesquelles il y a la possibilité d'apporter de bonnes contributions. Notre choix est tombé sur *le contexte sémantique* pour n'importe type de document multimédia, et *le contexte temporel* pour les vidéos. Il nous est paru logique d'étudier la possibilité d'utiliser les dépendances inter-plans de vidéos, puisque l'aspect temporel est indispensable pour comprendre le contenu d'une vidéo. D'autre part, nous avions plusieurs directions possibles concernant le contexte sémantique. Or, puisque notre but est la détection de n'importe quel type de concept, nous avons choisi d'utiliser les relations inter-concepts comme source d'informations contextuelles. Intuitivement,

ces choix paraissent adaptés aux concepts abstraits. En effet, si on prend par exemple les évènements, on peut les modéliser via d'autres concepts. Cette remarque reste valable pour n'importe quel type de concepts, puisqu'à la fin, un concept n'apparait jamais seul dans une image ou une vidéo. Par exemple, l'apparition d'un "ciel' implique implicitement l'occurrence d'autres concepts : "Dehors/En plein air", "Nuage", etc. En plus, les évènements durent plus ou moins longtemps et occurrent donc dans plusieurs plans successifs. C'est ce qui nous a motivé pour explorer ces pistes.

2.11.4 Comment utiliser le contexte ?

L'information contextuelle peut être intégrée à plusieurs niveaux possibles, dépendant du domaine abordé. La figure 2.11 présente cinq possibilités différentes d'utilisation du contexte correspondant chacune à une étape particulière d'un système d'indexation de documents multimédia :

1. Pré-traitement de données ;
2. Extraction et optimisation des descripteurs ;
3. Algorithme d'apprentissage/ Classification ;
4. Ré-ordonnancement de résultats de classification ;
5. Rétroaction ou boucle de pertinence ;

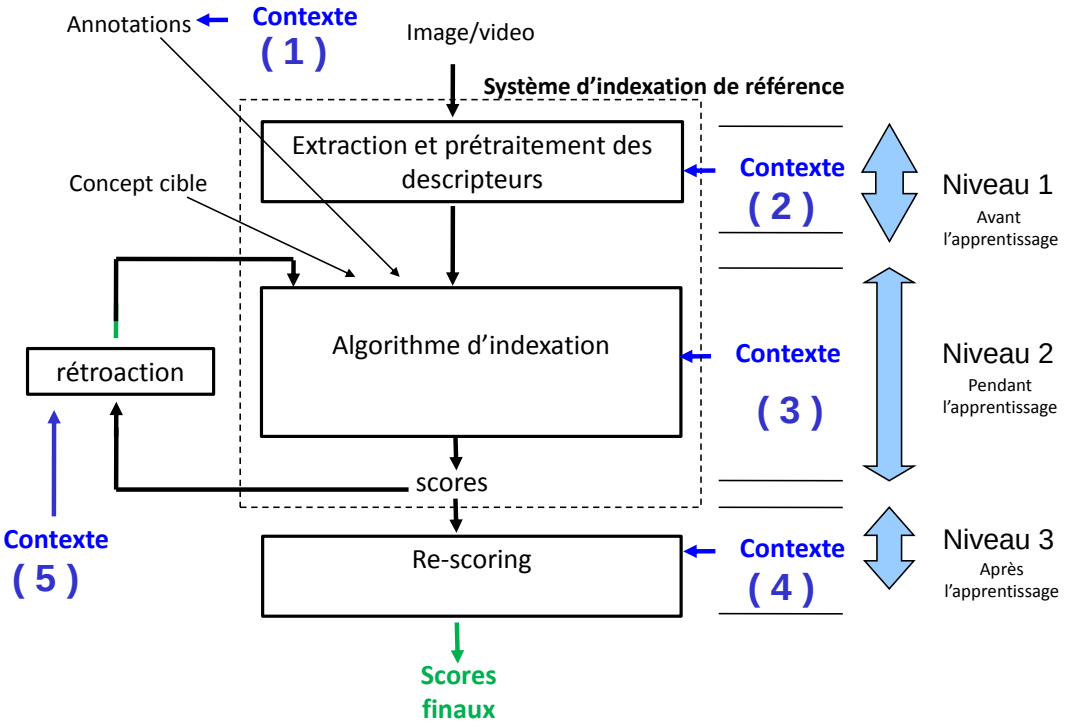


FIGURE 2.11 – Différentes possibilités d'utilisation du contexte dans un système d'indexation de documents multimédia.

Nous allons détailler dans ce qui suit chacune de ces cinq possibilités.

La première étape possible d'incorporation du contexte dans un système d'indexation est l'étape de pré-traitements de données. En effet, le contexte peut être considéré en pré-traitant les données avant le passage à l'étape d'apprentissage. À ce stade, ce sont les contextes environnemental et sémantique qui sont susceptibles d'être utilisés (voir la section 2.11.3). Par exemple, toute information qui entoure un document peut être considérée pour filtrer, pondérer et/ou nettoyer les annotations. Cela permettra d'avoir des données moins vulnérables au bruit qui peut influencer l'étape d'apprentissage. Le contexte sémantique peut être utilisé également à cette étape. Les relations entre les concepts en sont un exemple potentiel, comme dans [ZZY+13].

Le contexte peut être pris en compte dans l'étape d'extraction de descripteurs. Cela revient à utiliser un descripteur qui prend en compte des informations contextuelles. On peut parler du contexte spatial, dans lequel les différentes parties de l'image ou du plan vidéo sont distinguées, soit en divisant l'unité en plusieurs parties, typiquement quatre partie, pour extraire de chaque partie un descripteur et de combiner les différents descripteurs résultants. Ce processus peut être ré-itéré plusieurs fois comme proposé pour l'extraction de la pyramide spatiale [LSP06].

On peut considérer le contexte dans l'étape d'apprentissage. Cela peut se faire en utilisant par exemple, des noyaux prenant en compte des informations contextuelles, ou en développant une méthode spécifique qui utilise le contexte. Certains chercheurs ont privilégié l'utilisation de classificateurs discriminatifs pour intégrer l'information contextuelle, comme le boosting [WB06], SVM [QHR+07], ou des approches génératives comme "Naive Bayes" [KS03]. Certaines autres approches proposent d'utiliser les CRFs (Conditional Random Field) pour intégrer de l'information contextuelle à partir de différents niveaux d'information, à savoir le niveau "pixels" [SWRC09], le niveau "Objets" [RVG+07], l'image entière [MTF03] ou de multiples niveaux de l'image [Tor03].

La manière la plus simple d'utiliser le contexte est de le considérer dans une étape post-classification (i.e. ré-ordonnancement). Généralement on a tendance à fusionner les données contextuelles avec les sorties des classificateurs. Bien que cette approche présente l'avantage d'être simple et facile à mettre en œuvre, elle se heurte au problème de propagation des erreurs de l'étape de classification. Beaucoup de méthodes de l'état de l'art considérant le contexte sémantique appartiennent à cette catégorie de méthodes.

Une autre façon possible d'utiliser le contexte est "le mode rétroaction". Cette approche consiste à itérer l'étape d'apprentissage tout en introduisant de l'information contextuelle pour enrichir, corriger, ou appuyer les décisions des différents apprenants. Cela pourrait ressembler à une méthode de boucle de pertinence. Les nouvelles itérations d'apprentissage peuvent utiliser le même descripteur initial de bas niveau (dans le cas de correction des décisions des apprenants), ou un nouveau descripteur, typiquement de niveau sémantique ou de niveau intermédiaire, qui tient en compte le contexte. Une approche détaillée est présentée dans [HMQ13].

2.12 Détection simultanée de plusieurs concepts

Rechercher la co-occurrence d'un ensemble de concepts visuels dans des images/vidéos non annotées est une étape importante pour répondre à des requêtes complexes des utilisateurs, complexes en termes du nombre de termes composant la requête, ou en utilisant des termes très spécifiques ou génériques. En effet, ces requêtes sont souvent exprimées via un ensemble de termes sémantiques. D'autre part, considérer qu'un document multimédia est indexé par plusieurs concepts est utile : une simple combinaison d'un ensemble de concepts peut représenter d'autres sémantiques pouvant être complexes. Par exemple la combinaison des concepts "neige", "montagne" et "personne(s) en train de se déplacer" pourrait être liée à une scène de "skieur" ou "une compétition de ski". Cette problématique est liée à celle de l'utilisation du contexte. En effet, détecter simultanément un groupe de concepts revient à détecter un concept dans un contexte dans lequel les autres sont présents. Si la détection d'un seul concept est une tâche difficile, spécialement pour ceux qui sont rares ou difficiles à représenter ou décrire visuellement d'une manière efficace, cette difficulté s'accroit encore plus quand on veut vérifier l'occurrence conjointe de N concepts dans un même document. De plus, une scène, même avec uniquement deux concepts, tend à être complexe visuellement, et le défi reste difficile même dans le cadre d'une paire de concepts (bi-concepts). Cette remarque est confirmée par les résultats médiocres, en termes de performance, obtenus par les travaux qui se sont focalisés sur la détection de paire de concepts dans les images/vidéos. D'autre part, les approches de l'état de l'art concernant la détection de concepts dans les vidéos sont majoritairement basées sur l'apprentissage supervisé. Ces méthodes nécessitent des corpus de données dont leur obtention s'avère coûteuse en temps et argent. Donc, si on veut construire un modèle spécifique pour chaque paire de concepts, on se heurte forcément au problème du manque de données annotées. On distingue deux approches possibles pour détecter simultanément un groupe de concepts : 1) Considérer le groupe de concepts comme un nouveau concept et générer un modèle spécifique pour chaque groupe et 2) Détecter l'ensemble des concepts formant le groupe séparément et combiner ensuite les résultats de leurs détections respectives.

Dans l'état de l'art, les méthodes qui ont adressé le problème de détection simultanée de groupes de concepts n'ont pas été très convaincantes en matière de performance. Dans TRECVid 2012, une sous-tâche de la tâche d'indexation sémantique (SIN) a été introduite pour la première fois, pour traiter le problème de la détection de paires de concepts, à laquelle six équipes ont participé et le meilleur système a obtenu une valeur de précision moyenne (MAP, voir la section 2.13) qui ne dépasse pas les 8%, sachant que la performance des meilleurs systèmes pour détecter un seul concept dans la tâche SIN de TRECVid 2012 est de l'ordre de 30% en MAP. Cela montre l'étendu du défi concernant la problématique abordée.

Pour détecter simultanément un groupe de concepts, on pourrait combiner les résultats des détecteurs des concepts concernés. Même si la performance des détecteurs individuels est raisonnablement bonne, la fusion de leurs scores

donne de mauvais résultat, comme présenté dans [LSWS12] pour les paires de concepts. Un nombre considérable de travaux de recherche se sont focalisés sur cet axe. Dans [AHdVdJ08], les auteurs utilisent des règles de produit comme fonction de fusion, et plusieurs travaux [WJN11, SHH+07, LWLZ07, CHJ+06, YH03] ont étudié des fusion linéaires. Ces travaux ne concernent pas spécifiquement la détection simultanée de plusieurs concepts.

En ce qui concerne les études sur la détection de paires de concepts dans les documents multimédias, à savoir des images fixes, [LSWS12] proposent de générer un modèle par bi-concept à partir des échantillons collectés du web social. Les auteurs sont arrivés à la conclusion qu'apprendre des modèles de bi-concepts est mieux qu'une fusion linéaire de détecteurs de concepts individuels. La question reste cependant ouverte pour le cas des vidéos. Dans [WF09], les auteurs présentent une méthode pour apprendre des attributs visuels (p.ex, rouge, métal) et des classes d'objets (e.g. Voiture, robe, parapluie) ensemble, mais ce ne sont réellement pas des bi-concepts parce que le couple réfère à un et un seul et même concept (e.g. Le couple (voiture, rouge) fait référence à une même chose qui est une voiture).

Nous avons remarqué qu'il y a très peu d'études dans l'état de l'art qui ont adressé d'une manière spécifique le problème de détection simultanée de groupes de concepts dans les documents multimédia, encore moins pour le cas des vidéos.

Nous allons présenter dans le chapitre 6, une études détaillée qui compare les deux types d'approches de détection simultanée de groupes de concepts dans les images et vidéos.

2.13 Évaluation

2.13.1 Évaluation des systèmes d'indexation

Les annotations automatiques résultant d'un système d'indexation nécessitent une évaluation de leurs qualités. On a aussi besoin de comparer la performance d'un système d'indexation par rapport à certaines autres. Pour cela, plusieurs mesures d'évaluation ont été utilisées par les chercheurs. TREC a fourni un outil d'évaluation, nommé *Trec_eval*, qui permet de calculer plusieurs mesures de qualité de la détection d'un concept, comme par exemple : le rappel, la précision et la précision moyenne.

Les mesures d'évaluation les plus populaires, pour la comparaison des système de recherche d'information sont la précision et le rappel. Ces métriques sont largement utilisées pour l'évaluation de l'efficacité des approches d'annotation automatique, dans la communauté de la recherche d'information. Dans cette dernière, la précision d'une requête est définie par le ratio du nombre des documents pertinents retournés par le système et le total du nombre de documents retournés. Le le rappel quant à lui, est défini par le ration des documents pertinents retournés par le système et le nombre total des documents pertinents dans la base

de données.

$$précision = \frac{|\{documents\ pertinents\} \cap \{documents\ retournes\}|}{|\{documents\ retournes\}|} \tag{2.34}$$

$$rappel = \frac{|\{documents\ pertinents\} \cap \{documents\ retournes\}|}{|\{documents\ pertinents\}|} \tag{2.35}$$

Ces deux mesures ne considèrent pas toutes les informations indispensables à la comparaison des systèmes. Théoriquement, l'évaluation doit se baser sur des courbes montrant la précision comme une fonction du rappel. Cependant, il est aussi nécessaire d'inclure l'ordre dans lequel les documents sont retournés. Plusieurs autres mesures basées sur la précision et le rappel sont proposés :

P(10), P(30), P(N_r) : mesure la précision atteinte dans le top de la liste des $10, 30, N_r$ documents retournés.

R(10), R(30), R(N_r) : mesure la précision atteinte dans le top de la liste des $10, 30, N_r$ documents retournés.

Mean Average Precision (MAP) : mesure la précision moyenne non interpolée.

Inferred Average Precision (InfAP) : mesure la précision moyenne inférée.

Parmi ces mesures, la MAP et l'InfAP ont l'avantage de résumer la courbe "rappel-précision" en une seule valeur. Elles sont largement utilisées comme la mesure officielle de plusieurs compagnes de recherche d'images et de vidéos, comme TRECVid et Pacal-VOC. La MAP est définie par la formule suivante.

$$MAP = \frac{1}{R} \sum_{j=1}^{S} \frac{R_j}{j} \times I_j$$

où R est le nombre de plans de vidéos pertinents dans le un corpus contenant S plans. En considérant L la liste triée des documents retournés, à chaque indexe j, R_j est le le rappel après j plans qui sont retournés, et I_j est égal à 1 si le document document j est pertinent, et 0 sinon.

La mesure InfAP a été proposée par [YA06] pour être utilisée comme mesure d'évaluation dans TRECVid 2006.

2.13.2 Campagnes d'évaluation

2.13.2.1 Image clef

ImageCLEF vise à fournir un forum d'évaluation pour l'annotation interlangues et la recherche des images. Motivée par la nécessité de soutenir les utilisateurs multilingues d'une communauté mondiale accédant au corps d"information visuelle qui est croissant continuellement, l'objectif principal de la compagne ImageCLEF est de soutenir le progrès dans le domaine de l'analyse des médias visuels, l'indexation, la classification et la recherche, par le développement des

infrastructures nécessaires pour l'évaluation des systèmes de recherche d'information visuelle opérant à la fois sur des contextes monolingues, inter-langues et indépendants de la langue. ImageCLEF vise à fournir des ressources réutilisables pour ce genre de fins d'analyses comparatives.

ImageCLEF a été lancée en 2003 dans le cadre du Forum d'évaluation inter-Langues (CLEF :Cross Language Evaluation Forum), dans le but de fournir un soutien pour l'évaluation de : 1) méthodes indépendantes de la langue pour l'annotation automatique d'images par des concepts, 2) les méthodes multimodales de recherche d'information basées sur la combinaison de caractéristiques visuelles et textuelles, et 3) les méthodes de recherche d'images multilingues, afin de comparer l'effet de la recherche des annotations d'image et des formulations de la requête en plusieurs langues [6].

2.13.2.2 Pascal-VOC

PASCAL (Pattern Analysis, Statistical Modeling and Computational Learning) est un réseau [7] d'excellence financé par l'Union européenne. Il a mis en place un institut distribué qui réunit des chercheurs et des étudiants de toute l'Europe, et est maintenant ouvert à tous les pays du monde. PASCAL développe les résultats de l'expertise et scientifiques qui permettront de créer de nouvelles technologies telles que les interfaces intelligentes et des systèmes cognitifs adaptatifs. Pour ce faire, ce projet soutient et encourage la collaboration entre les experts en apprentissage automatique, statistique et optimisation. Il favorise également l'utilisation de l'apprentissage automatique dans de nombreux domaines d'application pertinents, tels que : le traitement des langages naturels, la recherche d'information, l'accès aux infirmations textuelles, ...

PASCAL Visual Object Classes (VOC) est un défi qui est devenu une référence dans le domaine de la détection et la reconnaissance des catégorie d'objets visuels, fournissant aux communautés de vision et d'apprentissage automatique un ensemble de données standard d'images, des annotations, et les procédures d'évaluation standards. Organisé chaque année depuis 2005 jusqu'à ce jour [8], ce défi été accepté comme une référence pour la détection des objets. Les données des différentes années sont disponibles en ligne sur le site officiel du défi [9]. Vous trouverez une description plus développée sur le défi, les données sont disponibles dans [EVGW+10].

2.13.2.3 TRECVid

Depuis 2001, la compagne d'évaluation TREC VIDEO offre à ses participant les moyens pour expérimenter différentes approches de détection de concepts dans les vidéos. Initié par l'Institut National de Standards et Technologies (NIST), la compagne TRECVid a visé de promouvoir les progrès scientifiques dans le

6. http ://www.imageclef.org/
7. http ://www.pascal-network.org/
8. Année de rédaction de la thèse : 2014
9. http ://pascallin.ecs.soton.ac.uk/challenges/VOC/

domaine de la recherche et l'indexation des vidéos. En mettant à dispositions des corpus de vidéos annotés et des outils d'évaluation ; TRECVid permet aussi aux participants de comparer leurs systèmes, chose qui en quelque sorte limite le travail exploratoire : comme elle est une part d'une compétition logique, tous les participants feront en sorte que leurs approches aient les meilleurs résultats.

Collection	Taille (heures)	Corpus de données	Concepts	Plans
2010	≈ 400	dev	130	118205
		test	30	144757
2011	≈ 600	dev	346	262962
		test	50	137327
2012	≈ 800	dev	346	400289
		test	50	145634
2013	≈ 1000	dev	346	545923
		test	60	112677

Table 2.3 – Les derniers corpus de données TRECVid.

Dans notre travail, nous avons utilisé les corpus de données des années allant de 2010 à 2013. Le tableau 2.3 résume l'évolution des collections TRECVid dans cette période. Comme on peut le voir, les ensembles de données ont été largement étendus, en termes de taille et du nombre de concepts, en atteignant 600 heures de vidéo en 2011, avec 130/346 concepts (130 pour 2010 et 346 concepts pour les autres années) annotés dans le corpus de développement et 50 concepts pour l'évaluation. Dans 2007-2009 TRECVid a mis à disposition aux participants, le magazine d'informations, documentaires et programmes d'éducation fournis par l'Institut néerlandais pour le son et la vision. En 2010 et 2011, TRECVid a fourni un nouveau corpus de vidéos, des communes créatives d'archive internet (Internet Archive Creative Commons : IACC [10]), caractérisé comme cela est courant dans beaucoup de "vidéos web" par la grande diversité de créateurs, contenu, style, qualités de production, l'encodage, langages, etc. Dans l'annexe A, le tableau A.2 présente la liste de tous les concepts (évalués ou non) TRECVid considérés dans les années 2010, 2011 et 2012, alors que le tableau A.1 présente la liste des concepts évalués dans les compagnes d'évaluation des années allant de 2009 et 2013.

Grâce à TRECVid, il est désormais possible d'évaluer les méthodes de traitement, d'analyse, de classification et de recherche d'information dans les grandes collections de vidéos. Les équipes de recherche à travers le monde, travaillent à des fins diverses sur des corpus de vidéos de l'ordre de plusieurs centaines d'heures. Toutes les expériences que nous présentons dans cette thèse ont été menées sur TRECVid corpus et de la tâche de détection concept (appelée aussi Extraction de descripteurs de haut niveau ou Indexation Sémantique). La compagne TRECVid est constituée de plusieurs tâches, on peut citer : la tâche d'indexation sémantique (SIN), la localisation, la vidéo surveillance, la détection et la reconnaissance des évènements, etc. Dans la cadre de cette thèse, nous avons travaillé uniquement dans la tâche SIN et sa sous-tâche : détection de paires de concepts.

10. http ://www.archive.org/index.php

La tâche d'indexation sémantique de TRECVid : La tâche d'indexation sémantique (SIN : Semantic INdexing) de TRECVid [OAF+10] est définie comme suit : étant donné un ensemble standard de plans pour la collection "test" pour l'indexation sémantique et une liste de définitions de concepts, les participants étaient invités à retourner pour chaque concept, tout au plus, les 2000 plans vidéo du jeu de test, classés par ordre décroissant par rapport à la probabilité, dans le sens la chance ou la possibilité, d'y détecter la présence du concept. La présence de chaque concept était supposée être binaire, c'est à dire, c'était soit présent ou absent dans le plan donné. Si le concept est présent dans certaines séquences du plan, alors qu'il est jugé de même pour le plan.

La tâche de détection de paires de concepts de TRECVid : Depuis 2012, une sous-tâche de la tâche d'indexation sémantique SIN de TRECVid, intitulée "détection de paires de concepts" a été mise en place. Elle consiste à détecter des paires de concepts non liés au lieu de la détection de concepts simples. Les deux concepts doivent apparaître dans au moins un plan pour que le plan soit considéré comme contenant la paire. L'idée est de promouvoir le développement de méthodes de recherche de plans vidéo contenant une combinaison de concepts qui font mieux que la fusion des sorties des détecteurs de concepts singuliers. À cette fin, chaque groupe participant est censé présenter une soumission de référence, qui combine juste pour chaque paire, la sortie des détecteurs des deux concepts singuliers formant la paire. L'objectif de la tâche est de voir si ce résultat de référence peut être dépassé ou pas.

2.14 Conclusion

Nous avons présenté dans ce chapitre une description d'un système classique d'indexation de documents multimédia basé sur la détection de concepts visuels, tout en détaillant chacune des étapes le composant et en faisant un tour sur les différentes approches possibles pouvant être considérées. Nous avons abordé ensuite, la notion du contexte en exposant ses différents types utilisés dans l'état de l'art. Nous avons finalement conclu par l'exposition des choix des approches et des catégories de contextes retenues dans le cadre de cette thèse. Nous présenterons dans les chapitres suivants nos contributions pour l'utilisation des contextes sémantique et temporel pour l'indexation des images et vidéos.

Chapitre 3

Contributions pour l'utilisation du contexte sémantique

Ce chapitre présente nos contributions concernant l'utilisation du contexte sémantique pour l'indexation sémantique des images/vidéos. Nous utilisons le contexte dans le but d'améliorer un système initial de détection de concepts dans des documents multimédia. Trois approches sont présentées et évaluées dans le contexte de la compagne TRECVid. Nous terminons ce chapitre par des conclusions et des perspectives pour améliorer nos propositions.

Nous présentons dans les sections suivantes nos propositions pour l'utilisation du contexte sémantique pour la détection de concepts visuels dans les images et vidéos. Nous rappelons que le contexte sémantique est restreint dans notre travail aux relations sémantiques entre concepts. Ces relations peuvent être issues de corpus de données, ou définies par un expert humain. Dans ce dernier cas, l'expert humain peut fournir différentes formes de sources de relations sémantiques. Ces ressources peuvent être des ontologies, ou une liste de relations explicites entre concepts, comme par exemple, les relations d'implication (e.g. Voiture $\Rightarrow$ Véhicule), des relations d'exclusion (e.g. Masculin *exclut* Féminin), etc.

Nous avons proposé trois approches qui utilisent le contexte sémantique. La première contribution nommée "Ré-ordonnancement sémantique basé sur une ontologie" utilise la hiérarchie d'une ontologie prédéfinie comme source de contexte, et se sert également des informations extraites d'un corpus de données pour les combiner dans le but de gagner en performance et robustesse. Dans la deuxième méthode que nous avons appelée "Reclassement sémantique par regroupement", nous modélisons le contexte en regroupant les exemples d'un ensemble d'apprentissage en se basant sur leurs contenus sémantiques détectés automatiquement par des détecteurs initiaux. La troisième proposition : "Rétroaction conceptuelle", consiste à construire un descripteur conceptuel de haut niveau et à l'utiliser de la même façon qu'un descripteur de bas niveau dans un système d'indexation sémantique. Pour la construction de ce descripteur, nous utilisons des sorties de détecteurs de concepts. Nous avons proposé deux extensions à "la rétroaction conceptuelle" : "Rétroaction conceptuelle étendue" et "Rétroaction conceptuelle itérative".

3.1 Notations

Nous allons considérer dans le reste de ce rapport les notations suivantes :
- C : est l'ensemble de concepts à détecter ;
- O : est une ontologie, c'est un ensemble de relations entre les concepts. Par exemple : les relations d'implication (Voiture $\Rightarrow$ Véhicule), les relations d'exclusion (Masculin *exclut* Féminin), etc ;
- E : est l'ensemble des échantillons considérés ;
- $E_D \subset E$: est l'ensemble des échantillons d'apprentissage manuellement annotés par au moins un concept de C ;
- $E_T = E \setminus E_D$: est l'ensemble des échantillons à annoter (échantillons de test) ;
- V : est un ensemble de vidéos, c'est une partition de E ;
- $F_{descr} : E \rightarrow R^n$, est une fonction qui donne une description d'un échantillon $e \in E$ où n est la taille du descripteur extrait ;
- $A : E_D \times C \rightarrow \{-1, 1\}$, est une fonction qui donne les annotations des échantillons d'apprentissage. $A(e, c) = 1$ signifie que le concept c est présent dans l'échantillon $e \in E_D$ et $A(e, c) = -1$ signifie que c est absent dans e ;
- $F_{sc} : E \times C \rightarrow R$, est une fonction de décision résultant d'un apprentissage indépendant pour chaque concept, sans utilisation du contexte ;
- $F_c : E \times C \rightarrow R$, est une fonction de décision dérivée de F_{sc} et basée sur le contexte. Cette fonction peut utiliser des informations externes comme une ontologie par exemple [WTS04] ;
- $F_{ac} : E \times C \rightarrow R$, est une fonction de décision finale fusionnant F_{sc} et F_c ;
- $F_v : E \rightarrow V$: est une fonction qui indique à quelle vidéo appartient un échantillon donné ;
- $F_p : E \rightarrow N$: est une fonction de numérotation des plans dans une vidéo. Pour chaque plan, F_p renvoie le rang dans lequel ce plan se trouve dans la vidéo à laquelle il appartient ;
- Une décision via une des fonctions F_{sc}, F_c ou F_{ac}, est un score de détection d'un concept dans un échantillon. Dans le cas des vidéos où l'ordre des plans est important, le score de détection d'un concept $c_i \in C$ dans un échantillon $e \in E$ est noté : $F_{sc}^{F_v(e),F_p(e)}(e, c_i)$, $F_c^{F_v(e),F_p(e)}(e, c_i)$ ou $F_{ac}^{F_v(e),F_p(e)}(e, c_i)$, au lieu de $F_{sc}(e, c_i)$, $F_c(e, c_i)$ ou $F_{ac}(e, c_i)$, pour le cas des documents fixes.

3.2 Ré-ordonnancement sémantique basé sur une ontologie

Plusieurs chercheurs exploitent les relations inter-concepts pour améliorer la performance des systèmes de détection de concepts. Dans ce travail, nous proposons de combiner l'utilisation d'une ontologie et d'un corpus. Nous présentons dans ce qui suit, deux approches de ré-ordonnancement qui considèrent la hiérarchie d'une ontologie prédéfinie. Nous allons comparer nos méthodes à une approche de l'état de l'art [WTS04].

En se basant sur les notations présentées dans la section 3.1, notre approche est modélisée par un 8-tuple :

$$< C, E(E_D \cup E_T), A, F_{descr}, O, F_{sc}, F_c, F_{ac} >$$

Notre proposition concerne la définition d'une méthode de ré-ordonnancement de résultats initiaux. Elle peut être appliquée quelque soit le système d'indexation utilisé. De ce fait, nous détaillons dans ce qui suit uniquement F_c et F_{ac}. Les détails des autres paramètres seront présentés dans la section 3.2.2 décrivant les expérimentations effectuées.

Pour détecter un concept $c \in C$, la première approche appelée *AncOrdesc* utilise les concepts ascendants et descendants de c dans la hiérarchie prédéfinie d'une ontologie de concepts. Un nouveau score est calculé en utilisant les scores de détection initiaux de c, de ses ascendants et de ses descendants dans l'ontologie O. Nous pondérons les scores de détection des ascendants et des descendants par leurs degrés de corrélation avec le concept cible c calculés sur un corpus de documents annotés manuellement.

La deuxième approche appelée *ConFamily* consiste en la spécification d'une famille pour chaque concept c à détecter. Une famille est un ensemble de concepts qui sont sémantiquement reliés au concept c et qui sont utiles pour sa détection. Pour détecter un concept c, nous proposons de mettre à jour le score de détection de c en le fusionnant avec ceux des membres de sa famille. La valeur de corrélation entre le concept c et les membres de sa famille est utilisée pour pondérer l'ensemble des concepts pris en compte.

3.2.1 Description de l'approche

Utiliser les relations entre concepts extraites de corpus de données est une bonne idée, mais le rôle d'un humain est aussi crucial. Une intervention humaine peut être substituée par l'utilisation d'une ontologie qui est elle même construite par un expert humain. Une simple utilisation de la hiérarchie d'une ontologie prédéfinie peut être critiquée pour plusieurs raisons. En effet, cela accorde des poids égaux à deux arcs reliant un concept c à deux autres concepts c_1 et c_2, alors que les degrés de similarité entre c *et* c_1 et c *et* c_2 sont différents. D'autre part, une ontologie ne considère pas les ensembles de données, ce qui ne la rend pas convenable à tous les types de données, conduisant à la nécessité de l'adapter aux données ou de la changer carrément. Pour pallier ces inconvénients, nous proposons :

1. D'utiliser une ontologie pour déterminer les concepts interdépendants ;
2. De ne pas utiliser les chemins reliant les concepts pour estimer la distance entre les concepts, mais de calculer des poids à base de corpus de données, et les utiliser comme mesures de similarités. Cela permet d'utiliser simultanément les informations de l'ontologie et celles d'un ensemble de données. Inspiré par [ZWLX10], nous utilisons les coefficients de corrélation comme

poids :

$$Corr(c_i, c_j) = \frac{\sum_{e \in E_D} (A(e, c_i) - \bar{A(., c_i)}) \times (A(e, c_j) - \bar{A(., c_j)})}{\sqrt{\sum_{e \in E_D} (A(e, c_i) - \bar{A(., c_i)})^2} \times \sqrt{\sum_{e \in E_D} (A(e, c_j) - \bar{A(., c_j)})^2}} \tag{3.1}$$

où
$\bar{A(., c_k)}$ représente la moyenne des valeurs $A(e, c_k)$: $\bar{A(., c_k)} = \sum_{e \in E_D} A(e, c_k)/||E_D||$

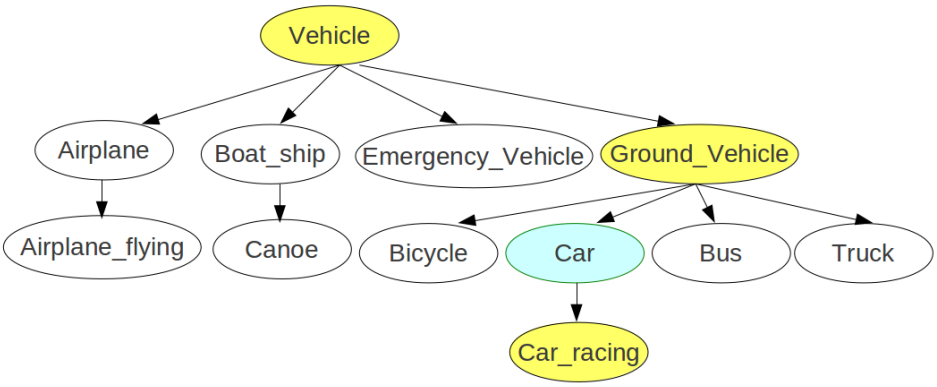


FIGURE 3.1 – Sélection des ancêtres et descendants du concept cible dans la hiérarchie de concepts (e.g. Car).

Pour détecter un concept $c_i \in C$ dans un échantillon $e \in E_T$, un score initial d'une première classification $F_{sc}(e, c_i)$ est modifié en exploitant les scores de détection d'un ensemble d'autres concepts et le degré de corrélation entre ces concepts et c_i, pour générer un score $F_c(e, c_i)$ basé sur le contexte. Finalement, un nouveau score $F_{ac}(e, c_i)$ est ensuite utilisé pour ordonner les échantillons en termes de pertinence par rapport au concept cible c_i.

Nous proposons dans ce qui suit deux approches pour déterminer les concepts à utiliser pour améliorer la détection de $c_i \in C$ dans un échantillon $e \in E_T$:

1. "Ancêtres et/ou descendants" : pour un concept c_i, combiner uniquement les concepts qui sont des ancêtres ou des descendants de c_i dans la hiérarchie de l'ontologie. La figure 3.1 montre un exemple d'une telle sélection pour concept "Car". Le nouveau score de détection de c_i dans e ($F_{ac}(e, c_i)$) est donné par la formule suivante :

$$F_{ac}(e, c_i) = F_{sc}(e, c_i) + F_c(e, c_i) \tag{3.2}$$

où :

$$F_c(e, c_i) = \sum_{c_j \in ont(c_i)} Corr(c_i, c_j) \times F_{sc}(e, c_j) \tag{3.3}$$

avec $ont(c_i) = \{c_j \in C |\ c_j$ est un ancêtre ou un descendant de c_i dans l'ontologie$\}$;

2. "Famille de concept" : Pour un concept c_i, un expert humain sélectionne les concepts reliés au concept c_i. D'après nos expérimentations, nous avons trouvé que les co-occurrences des concepts montrent que pour un concept c_i les concepts qui sont ancêtres, descendants dans l'ontologie, comme certains autres concepts qui partagent un même ancêtre de c_i co-occurent souvent avec c_i. D'autres expérimentations ont montré que ce n'est pas nécessairement l'ensemble de ces concepts qui aide à détecter c_i. Nous proposons donc qu'un expert humain sélectionne pour chaque concept c_i un ensemble de concepts représentant ce que nous appellerons : "famille de concept". Une famille du concept c_i est un ensemble de concepts reliés sémantiquement à c_i. Les relations entre les concepts induisent une structure en treillis dans l'ensemble de concepts. Au sein de cette structure, un expert humain peut identifier et extraire un certain nombre d'arbres ou des fragments, chacun représentant une hiérarchie de concepts compatible avec un sens commun. Les fragments résultants ne sont pas nécessairement déconnectés. En effet, les arbres concernant certains concepts peuvent se chevaucher, comme par exemple pour les deux concepts "voiture" et "Sport" dont leurs fragments respectifs pourraient contenir des concepts en commun concernant par exemple les sports impliquant l'utilisant d'une voiture (e.g. "Car_Racing"). L'union des fragments de tous les concepts forme le corps entier de l'ontologie de départ. Ensuite, une famille initiale d'un concept c_i est définie comme l'ensemble contenant tous les descendants de chaque ancêtre de c_i (toute l'arborescence à laquelle c_i appartient). Après avoir sélectionné cette famille initiale, et en se basant sur un corpus de développement (par validation croisée), on élimine les concepts qui n'aident pas à améliorer la détection de c_i. Par exemple, comme montré dans la figure 3.2, la famille du concept "Car" contiendra : *Vehicle*, *Emergency_Vehicle*, *Ground_Vehicle*, *Airplane*, *Airplane_Flying*, *Bicycle*, *Bus*, *Truck*, *Car* et *Car_ Racing*. De même, "Airplane_Flying" aura dans sa famille : *Airplane*, *Airplane_Flying* et *Vehicle*. Le nouveau score de détection de c_i dans un échantillon e ($F_{ac}(e, c_i)$) est donné par la formule suivante :

$$F_{ac}(e, c_i) = F_{sc}(e, c_i) + F_c(e, c_i) \tag{3.4}$$

où :

$$F_c(e, c_i) = \sum_{c_j \in Fam(c_i)} Corr(c_i, c_j) \times F_{sc}(e, c_j) \tag{3.5}$$

avec $Fam(c_i) = \{$ tous les concepts jugés sémantiquement reliés à c_i par un expert humain $\}$;

3.2.2 Expérimentations et résultats

Nous avons testé et évalué les approches "Ancêtres et/ou descendants" et "Famille de concept" dans le contexte de la tâche d'indexation sémantique de TRECVid 2010. L'étape de ré-ordonnancement dépend des scores de détection de concepts fournis par des détecteurs individuels de concepts. Nous avons choisi

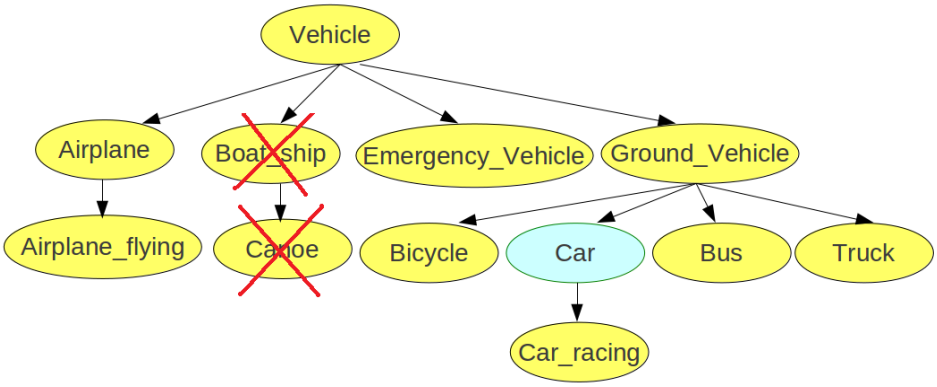


FIGURE 3.2 – Sélection de la famille d'un concept cible (e.g. Car).

d'utiliser comme classificateurs, MSVM [SQ10] (voir la section 2.7.4.2) et une variante de KNN [YH08a] : KNNC [1], pour leurs bons résultats dans le cadre de l'indexation de vidéos. Comme entrée de ces classificateurs, cinq descripteurs ont été extraits pour chaque plan vidéo, et utilisés comme des vecteurs caractéristiques. Nous avons utilisé des descripteurs de couleurs, de texture, des descripteurs basés sur les points d'intérêts (SIFT) pour la modalité visuelle, MFCC pour l'audio, comme décrit ci-après. Ces descripteurs ont été extraits par les équipes LIG [2] et GIPSA [3] du groupe IRIM [4]. Voici quelques détails supplémentaires :

- LIG/$h3d64$ ($hist$) : histogramme RGB normalisé 4×4×4 (64-dim) ;
- LIG/$gab40$ (gab) : transformation de Gabor normalisée, 8 orientations × 5 échelles (40-dim) ;
- LIG/$hg104$ (hg) : fusion précoce (concaténation) de LIG/$h3d64$ et LIG/gab40 (104-dim) ;
- LIG/$opp_sift_har_1000$ ($sift$) : sac de mots sur des descripteurs SIFT (opponent sift) avec le détecteur Harris-Laplace (1000-dim) ;
- GIPSA/$AudioSpectro.b28$ ($audio$) : profil spectral en 28 bandes sur l'échelle de Mel ;

Voir l'annexe B pour plus de détails sur ces descripteurs.

Nous avons fait ce choix de descripteurs simples, parce que nous nous focalisons dans ce travail sur l'étape de re-ordonnancement. Cependant, pour tester la robustesse de nos approches, nous avons fait une fusion tardive des résultats de 40 descripteurs individuels, afin de tester notre approche sur un système initial assez bon en termes de MAP. Nous avons comparé nos propositions avec l'approche "facteurs de boosting et de confusion" détaillée dans [WTS04], dans laquelle une ontologie et des relations sémantiques de type "c_i exclut c_j" ont été utilisées dans une étape de re-ordonnancement.

1. KNNC : est une version de KNN dans laquelle les paramètres sont optimisés séparément pour chaque concept
2. http ://www.liglab.fr/
3. http ://www.gipsa-lab.inpg.fr/
4. http ://mrim.imag.fr/irim/

Notre évaluation a été menée sur la collection TRECVid 2010. Nous avons déroulé nos expérimentations sur un corpus de développement divisée en deux parties : Une pour l'apprentissage et l'autre pour l'évaluation ("validation croisée 1-fold"). Les annotations ont été fournies par l'annotation collaborative TRECVid 2010 organisée par les équipes LIG et LIF [AQ08a]. Nous avons utilisé un lexique contenant 130 concepts. L'ontologie utilisée dans nos expérimentations a été construite sur la base de relations inter-concepts de type $c_1 \Rightarrow c_2$, de la manière suivante : si $c_1 \Rightarrow c_2$ alors c_2 est un ancêtre de c_1. La performance du système a été mesurée par la précision moyenne (MAP) calculée sur les 130 concepts. Les annotations fournies ne sont pas indépendantes de l'ontologie. En effet, une fermeture transitive à été appliquée aux annotations brutes. Cela est un désavantage pour notre approche, puisque nous allons tenter d'améliorer un système qui a exploite déjà du contexte sémantique.

Nous avons effectué les expérimentations suivantes :

1. Lancer MSVM et KNNC pour chaque concept en utilisant les différents descripteurs cités précédemment, puis appliquer les approches de ré-ordonnancement ;
2. Fusion tardive des scores de MSVM et KNNC pour chaque descripteur, et application du ré-ordonnancement après ;
3. Fusion tardive des résultats obtenus par les détecteurs de concepts individuels correspondant aux descripteurs (Nous notons ce résultat : *fusion_desc*), et application du ré-ordonnancement après ;
4. Pour tester la robustesse de nos approches, nous les avons appliquées sur les résultats d'un système initial avec une bonne performance en termes de MAP, qui est obtenu par une fusion tardive des résultats de 40 descripteurs (couleurs, textures, audio, sift, ...). La valeur de MAP calculée sur les résultats de ce système dépassent 14%. Nous notons ce résultat : *Quaero_fusion*. Notons que dans TRECVid 2010, le meilleur système a eu une valeur de MAP d'environ 9% [2]. Nous avons appliqué le ré-ordonnancement en utilisant les scores *Quaero_fusion* ;

Nous utiliserons dans ce qui suit les notations suivantes : 1) *BoostingF* : l'approche "Boosting factor", 2) $ConfusionF$: l'approche "confusion factor" , 3) *concFamily* : l'approche "famille de concept", 4) *AncOrdesc* : l'approche "Ancêtres ou descendants".

Résultats

Le tableau 3.1 présente une comparaison entre les résultats obtenus par nos approches et ceux de deux méthodes de l'état de l'art : "Boosting factor" et "Confusion factor". Ces résultats concerne le ré-ordonnancement sur les résultats de la fusion tardive des scores de KNNC et MSVM. On remarque que $ConfusionF$ détériore toujours les résultats, alors que la méthode *BoostingF* se comporte mieux avec les descripteurs individuels, mais l'amélioration est moins importante quand le système initial est plus performant. D'autre part, nos propositions

2. Dans TRECVid 2010, la MAP est calculée sur uniquement 30 concepts

		État de l'art		Nos approches	
	initial	*BoostingF*	*ConfusionF*	*AncOrdesc*	*ConFamily*
hist	0.0343	0.0345 (+0.58)	0.0341 (-0.58)	0.0347 (+1.17)	0.0356 (+3.79)
gab	0.0307	0.0309 (+0.65)	0.0306 (-0.32)	0.0311 (+1.30)	0.0315 (+2.60)
hg	0.0548	0.0549 (+0.18)	0.0546 (-0.36)	0.0550 (+0.36)	0.0559 (+2.01)
sift	0.0698	0.0710 (+1.72)	0.0696 (-0.28)	0.0711 (+1.86)	0.0725 (+3.87)
audio	0.0136	0.0138 (+1.47)	0.0135 (-0.73)	0.0142 (+4.41)	0.0146 (+7.35)
fusion_desc	0.0832	0.0844 (+1.44)	0.0827 (-0.60)	0.0841 (+1.08)	0.0856 (+2.88)
Quaero_fusion	0.1428	**0.1447** (+1.33)	**0.1419** (-0.63)	**0.1457** (+2.03)	**0.1478** (+3.50)

TABLE 3.1 – Résultats (MAP (gain %)) pour nos deux différentes approches et une comparaison avec ceux de méthodes de l'état de l'art.

améliorent les résultats quelque soit le type de descripteur utilisé et quelque soit la performance du système, et le gain est meilleur que celui atteint par la méthode "BoostingF". En outre, "famille de concept" donne de meilleurs résultats que l'approche "Ancêtre et descendants ", atteignant un gain qui varie entre +2.01% et +7.35%. La meilleure valeur de MAP d'environ 14.78% a été obtenue en utilisant *Quaero_fusion*. Même si cette valeur de MAP semble basse, un système d'indexation reste bon et utilisable, surtout qu'il s'agit d'une précision moyenne et qu'au début de la liste la précision est plus élevée.

En appliquant *ConFamily* sur les scores de chaque classificateur séparément, le gain varie entre 2.92% et +7.88% pour KNNC et entre +0.88% et +7.02% pour MSVM.

Nous rappelons que *ConFamily* fusionne des scores obtenus par des classificateurs entrainés indépendamment. Ainsi, on se heurte au problème de normalisation des sorties de classificateurs. Nous pensons que prévoir une étape de normalisation des scores avant l'application de cette méthode permettrait d'avoir des améliorations plus importantes.

3.3 Reclassement sémantique par regroupement

L'approche que nous proposons et que nous appelons par la suite *reclassement sémantique par regroupement*, s'inspire de [SGG+06]. Elle est basée sur le constat suivant : en raison de la richesse du contenu d'une image/vidéo en termes de sémantiques, tenter de détecter un concept visuel seul est une idée très naïve. En effet, les concepts n'existent pas isolément, certains concepts co-

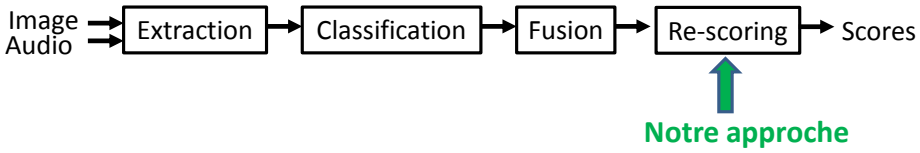


FIGURE 3.3 – Positionnement de l'approche "reclassement sémantique par regroupement" dans un système d'indexation de documents multimédia.

occurrent toujours (*Animal* & *Véhicule*), certains autres très souvent (*Ciel* & *Avion*). D'autre part, la présence de certains concepts exclut l'occurrence de certains autres (*Féminin* & *Masculin*). Nous supposons qu'utiliser des informations liées à d'autres concepts permettrait d'améliorer les performances initiales de détection d'un concept cible, par rapport au cas où aucune autre information externe n'est utilisée. Sur la base de cette idée, si l'on considère que nous avons un score de détection pour chaque concept, et en considérant pour chaque échantillon les scores de détection d'un ensemble de concepts comme vecteur caractéristique, nous pensons que les échantillons positifs se réuniront dans l'espace, et un échantillon négatif sera plus éloigné (qu'un exemple positif) des centres des groupes qui contiennent beaucoup d'exemples positifs. Cela conduit à notre proposition.

3.3.1 Description de l'approche

Nous présentons dans ce qui suit, une modélisation générale du système proposé pour l'indexation de documents multimédia par détection de concepts. Ce modèle se base sur les annotations manuelles d'un ensemble de documents multimédia et certaines fonctions permettant de générer de nouveaux scores de détection.

Par rapport aux notations présentées dans la section 3.1, nous proposons de modéliser notre approche par un 6-tuple :

$$< C, E(E_D \cup E_T), A, F_{sc}, F_c, F_{ac} >$$

Instanciation du modèle

Notre approche proposée : "reclassement sémantique par regroupement" s'inscrit dans une étape de re-scoring comme montré dans la figure 3.3. Notre approche vise à exploiter les scores de détection d'un grand nombre de concepts. Ces scores sont obtenus par des détecteurs de concepts basés sur un apprentissage automatique sur l'ensemble E_D. Plus le score est grand, plus la probabilité que les concepts soient présents est grande. Pour détecter un concept $c_i \in C$ dans un échantillon $e \in E_T$, un score initial d'une première classification $F_{sc}(e, c_i)$ est modifié en exploitant les scores de détection d'un ensemble d'autres concepts. Il en résulte un score final $F_{ac}(e, c_i)$ de détection du concept cible c_i dans e.

Plus précisément, la détection d'un concept c_i dans un document multimédia $e_t \in E_T$ est décrite par les étapes suivantes :

1. calculer $F_{sc}(e, c_i)$ pour tout $c_i \in C$, et $e_d \in E_D$ ainsi que pour e_t. Ces valeurs ont un impact majeur sur tout le traitement. Nous notons $F_{sc}(e, .)$ le vecteur contenant les scores de détection de tous les concepts appartenant à C, dans l'échantillon e ;
2. Faire un regroupement (clustering) en se basant sur $F_{sc}(e_d, .)$ de tous les exemples $e_d \in E_D$ annotés positivement ou négativement par le concept c_i. Tous les échantillons, y compris e_t, seront représentés donc par de nouveaux descripteurs de haut niveau, $F_{sc}(e, .)$, et un regroupement des échantillons est réalisé à base de ces nouveaux descripteurs. Le résultat est un ensemble de CL groupes (clusters) $clus_j^{c_i}$ où $j \in [1, CL]$. L'idée derrière un tel regroupement est d'être en mesure de considérer implicitement les relations entre les concepts, en utilisant les scores des autres concepts. N'importe quelle technique de regroupement peut être utilisée à ce stade, comme par exemple *K-means*, que nous avons choisie dans notre cas, pour sa simplicité. Le nombre de groupes (clusters) est un paramètre à optimiser sur un corpus de développement ;
3. Les groupes définis précédemment contiennent des documents annotés positivement et négativement par le concept c_i. Nous pouvons voir ces groupes comme étant des contextes différents reliés au concept c_i. En effet, chaque groupe est censé contenir un ensemble d'échantillons ayant des contenus sémantiques similaires (proches). Nous estimons la probabilité des exemples positifs dans chaque groupe $clus_j^{c_i}$, et nous attribuons à chacun de ces groupes un indicateur de sa pertinence par rapport à c_i. Cette qualité est estimée par la probabilité $P(+/clus_j^{c_i})$ d'apparition de c_i dans chaque groupe :

$$P(+/clus_j^{c_i}) = \frac{\#(+, clus_j^{c_i})}{\#(+-, clus_j^{c_i})} \tag{3.6}$$

   avec : $\#(+, clus_j^{c_i})$ est le nombre d'exemples dans $clus_j^{c_i}$ qui sont manuellement annotés positifs par c_i, et $\#(+-, clus_j^{c_i})$ est le nombre de tous les échantillons dans $clus_j^{c_i}$.
4. Nous proposons ensuite, de calculer la distance séparant $F_{sc}(e_t, .)$ des centres des différents groupes $clus_j^{c_i}$, que nous notons $dist(e_t, clus_j^{c_i})$. L'étape suivante consiste à définir la valeur $F_c(e_t, c_i)$. Cette fonction nécessite la définition d'une mesure de similarité entre l'échantillon à annoter et les différents groupes. Nous considérons la méthode des K-plus proches voisins (KPPV ou KNN en Anglais), où K dénote le nombre des groupes voisins considéré :

$$F_c(e_t, c_i) = \frac{\sum\limits_{j \in K \text{ plus proches groupe.}} \frac{P(+/clus_j^{c_i})}{dist(e_t, clus_j^{c_i})}}{\sum\limits_{j \in K \text{ plus proches groupe.}} \frac{1}{dist(e_t, clus_j^{c_i})}} \tag{3.7}$$

5. La dernière étape consiste à fusionner les scores $F_c(e_t, c_i)$ et $F_{sc}(e_t, c_i)$ pour déduire un score final de détection $F_{ac}(e_t, c_i)$. Nous avons testé plusieurs

possibilités et nous avons retenu pour nos expérimentations *la combinaison linéaire pondérée* :

$$F_{ac}(e_t, c_i) = \alpha \times F_{sc}(e_t, c_i) + (1 - \alpha) \times F_c(e_t, c_i) \tag{3.8}$$

où α est un paramètre réel appartenant à l'intervalle [0, 1] qui dénote l'influence relative du contexte sur l'occurrence du concept.

Notre système modifie donc, les annotations automatiques initiales pour un concept cible c_i, en intégrant des annotations des exemples d'un ensemble d'apprentissage par rapport à c_i et les scores de détection automatique d'autres concepts. Notre proposition ne présuppose aucune caractéristique spécifique concernant les scores initiaux ($\in [0, 1]$), qui peuvent être calculés en utilisant n'importe quelle approche possible de détection de concepts.

Les paramètres caractérisant notre proposition sont :

- Le nombre de groupes (clusters) CL pour chaque concept $c_i \in C$;
- La distance utilisée pour le calcul de $dist(e_t, clus_j^{c_i})$. Dans les expérimentations, nous avons utilisé pour sa simplicité, la distance de Manhattan $L1$, normalisée par le nombre de dimensions ($|C|$), pour contrôler la dispersion des valeurs ;
- Le paramètre K qui dénote le nombre de voisins considérés pour le calcul de F_c ;
- Le paramètre $\alpha \in [0, 1]$ défini dans la fonction F_{ac}, qui reflète l'importance relative des scores initiaux et du contexte.

3.3.2 Expérimentations et résultats

Données :

Notre évaluation a été réalisée sur les collections de données TRECVid 2012. Nous avons considéré un lexique de 346 concepts. Nos expérimentations ont été faites sur trois corpus : apprentissage, validation et test. Les annotations ont été fournies par l'annotation collaborative TRECVid 2012 [AQ08a]. Nous avons utilisé la précision moyenne (MAP) comme mesure de performance, calculée sur les 346 concepts pour le corpus de validation, et sur 46 concepts pour le corpus de test. Cela est imposé à la procédure *officielle* d'évaluation de TRECVid, et aussi parce que nous ne disposions pas des annotations des échantillons par tous les concepts.

Expérimentations :

Nous avons utilisés 20 types de descripteurs. Ces descripteurs portent sur les textures/couleurs, SIFT, STIP, VLAD, VLAT, Percepts, etc. 100 variantes au total de ces descripteurs (Par exemple, en variant la taille du dictionnaire pour les "bags of word") ont été considérées. L'ensemble des descripteurs utilisé est décrit dans [BLS+12] ou dans l'annexe B. Nous pouvons diviser nos expérimentations en trois étapes :

a) Détection initiale des concepts :
En raison de leurs bons résultats, nous avons choisi d'utiliser MSVM [SQ10] et KNN [YH08a] comme classificateurs supervisés. Comme entrée de ces détecteurs, les descripteurs décrits ci-dessus ont été utilisés. Pour chaque paire (concept, descripteur), les expérimentations suivantes ont été effectuées : 1) Apprentissage et optimisation des paramètres sur l'ensemble de développement, 2) Prédiction sur le corpus de validation et évaluation sur 346 concepts et 3) Prédiction sur le corpus de test et évaluation sur 46 concepts.

b) Fusion :
Une fusion tardive des scores obtenus dans l'étape de détection initiale des concepts est appliquée dans le but d'améliorer les performances. Cela est réalisé en fusionnant pour chaque exemple, un nombre de scores obtenus par les différents descripteurs. Nous avons testé trois types de fusion :

- $Fusion_1$: est une fusion hiérarchique, qui est décrite dans [SDH+12] (voir la section 2.8.2) ;
- $Fusion_2$: fusion basée sur MSVM, en utilisant en entrée les résultats de $Fusion_1$. C'est une approche de stacking (voir la section 2.7.4.4) basée sur les MSVM ;
- $Fusion_3$: fusion des résultats de $Fusion_1$ et $Fusion_2$ (moyenne) ;

$Fusion_1$, $Fusion_2$ et $Fusion_3$ donnent de bons résultats. En effet, leurs valeurs de MAP sont supérieures à 20%, ce qui donnerait un bon classement dans l'évaluation officielle de TRECVid. Nous avons utilisé ces trois résultats de fusion dans l'approche de reclassement.

c) Reclassement sémantique par regroupement :
En plus des paramètres des classificateurs, nous avons optimisé pour chaque paire (concept, descripteur), le nombre de groupes (CL) pour *k-means* et α pour la fonction de fusion F_{ac}. Dans nos expérimentations, nous avons testé une optimisation globale de ces paramètres, ce qui a mené à des résultats médiocres en termes de MAP. Cela peut être expliqué d'une part, par la différence entre le nombre et des instances des exemples positifs et négatifs entre les concepts, et d'autre part, par la différence entre les résultats atteints par les différents descripteurs. En effet, utiliser différents descripteurs donne différentes distributions de scores et donc, différentes représentations des exemples, ce qui conduit à un regroupement différent des échantillons. Par conséquent, nous avons opté pour une optimisation locale pour chaque paire (concept,descripteur), un choix à base duquel les résultats suivants sont obtenus.

Résultats et discussion

Le tableau 3.2 montre les résultats de notre proposition sur les corpus de validation et de test.

Notre approche proposée améliore les résultats sur le corpus de validation, quelques soient les résultats initiaux utilisés en entrée. Le gain relatif varie entre

	Corpus de validation		Corpus de test	
	MAP init	MAP (gain) après reclassement	MAP init	MAP (gain) après reclassement
$Fusion_1$	0.2469	0.2525 (+2.27%)	0.2600	0.2591 (-0.34%)
$Fusion_2$	0.2010	0.2139 (+6.42%)	0.2431	0.2522 (+3.75%)
$Fusion_3$	0.2488	0.2538 (+2.01%)	0.2749	0.2774 (+0.90%)

TABLE 3.2 – Résultats après reclassement par regroupement.

+2.01% et +6.42%. Cette différence de gain peut être expliquée par la différence entre les performances des détecteurs initiaux : il est plus difficile d'améliorer un système qui est déjà bon qu'un mauvais système. Nous avons utilisé le test de *Student* pour tester la significativité statistique des différences entre les précisions moyennes obtenues. Pour les résultats sur le corpus de validation, toutes les améliorations sont hautement significatives, donnant des valeurs de p inférieures à 3.1E-14.

Pour la collection de test, le reclassement par regroupement sémantique améliore les résultats sauf pour $Fusion_1$. Le gain est significatif (p = 3.75 %) pour $Fusion_2$, mais pas pour les deux autres. Cela peut être expliqué par la procédure d'évaluation officielle de TRECVid qui considère uniquement 46 concepts. Cependant, nous avons obtenu de meilleurs résultats sur le corpus de validation où 346 concepts ont été utilisés.

La figure 3.4 montre la variation des précisions moyennes (AP) en fonction du paramètre α, pour certains concepts représentatifs : *sciences_technology*, *old_people*, *Sky*, *Overlaid_text* et *Actor*. Nous pouvons remarquer que les scores basés sur le contexte seuls (F_c, donc $\alpha = 0$) ne permettent pas une meilleure détection de concepts, comparés aux détecteurs initiaux ($\alpha = 1$), sauf pour le concept *Science _technology*. D'autre part, les valeurs de α maximisant la précision moyenne AP sont généralement inférieures à 0.5, ce qui signifie que le poids des scores F_c est plus important que celui des scores initiaux F_{sc}, cela peut être expliqué par le fait que F_c fournissent de l'information utile pour la correction du classement résultant de F_{sc}. Ce phénomène permet également de contourner le problème de "normalisation des scores" (voir la section 2.7.6), où les scores initiaux écrasent les nouveaux scores, ou inversement : les valeurs de α servent également à uniformiser les deux distributions de scores, en les remettant sur une même échelle.

Nous avons trouvé aussi que le nombre de groupes (clusters) affecte la performance, ce qui signifie que c'est mieux de l'optimiser localement pour chaque concept. D'autre part, plus les valeurs des paramètres CL et K augmentent, plus la précision moyenne est meilleure.

Malgré que notre évaluation a porté sur des corpus vidéo, elle reste applicable sur n'importe quel type de documents multimédia, notamment les images.

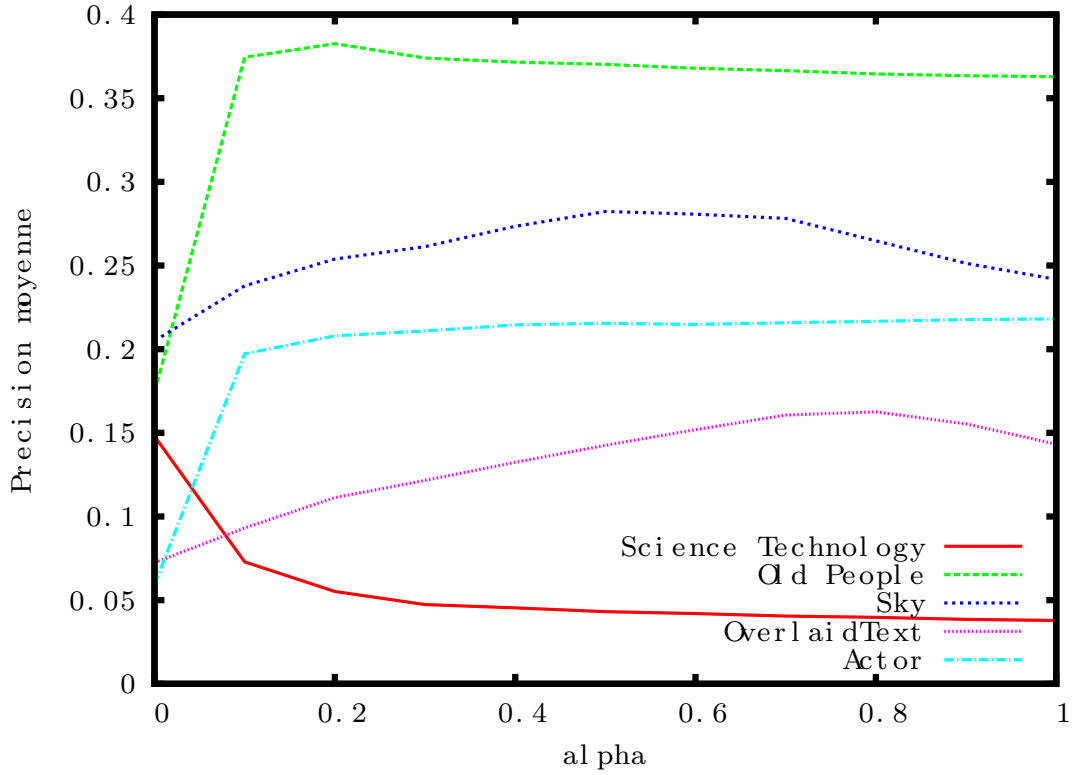


FIGURE 3.4 – Variations de la précision moyenne en fonction du paramètre α. Ces résultats sont obtenus en considérant : $CL = 200$, $K = 50$ pour les cinq concepts.

3.4 Rétroaction conceptuelle

Le pipeline classique "extraction/classification/fusion" a été appliqué avec succès pour l'indexation sémantique de documents multimédia, mais la performance globale reste faible. En effet, la précision moyenne (MAP) de l'état de l'art des systèmes d'indexation dans TRECVid varie dans l'intervalle [0.1, 0.3] [SOK09]. Dans la compagne d'évaluation TRECVid 2012, le meilleur système a atteint une MAP d'environ 32%. Au cours des dernières années, la performance a été progressivement améliorée grâce à l'utilisation de meilleurs descripteurs, plus de descripteurs, de meilleures méthodes de classification, de meilleurs systèmes de fusion et une meilleure annotation. Cette approche classique peut encore être améliorée en tenant compte de la cohérence temporelle des contenus de vidéo et/ou les relations entre les concepts cibles quand un certain nombre d'entre eux doivent être détectés simultanément. Ces aspects ont été pris en compte dans plusieurs travaux récents et l'approche présentée dans ce travail vise également à les exploiter pour améliorer la performance d'un système d'indexation sémantique initial.

La méthode que nous proposons dans cette partie considère principalement la dimension conceptuelle. Son originalité vient du fait qu'il s'agit d'une rétroaction des scores de détection via un descripteur additionnel, qui est similaire aux "vecteurs modèles' proposés par Smith et al. [SNN03]. Ce descripteur de haut niveau

passe aussi par les étapes de classification et de fusion. La dimension temporelle peut être aussi prise en compte ou bien d'une manière séquentielle ou une approche alternative.

Cette étude détaillée qui tient compte à la fois des contextes sémantique et temporel à la fois, est à notre connaissance la première sur les données TRECVid 2012. Un des points forts de notre travail est le fait que nous considérons un système de référence déjà assez performant, avec une performance comparable à celle que nous avons soumis à l'évaluation dans la tâche d'indexation sémantique de TRECVid 2012, et qui nous a permis d'obtenir un bon classement.

3.4.1 Description de l'approche

Nous considérons un système de référence suivant le pipeline classique de type "extraction/classification/fusion" comme montré dans la figure 3.5. n_e méthodes sont utilisées pour extraire des descripteurs de l'image ou des pistes audio des échantillons vidéos (typiquement des plans). $n_e \times n_c$ *classificateurs* sont ensuite entraînés séparément pour chaque couple (descripteur, concept), où n_c est le nombre de concepts qu'on veut détecter. n_c modules de fusion *(tardive)* sont finalement utilisés pour produire n_c scores représentant la probabilité qu'un échantillon vidéo contiennent les concepts cibles. Le même ensemble de descripteurs et le même schéma de fusion sont utilisés pour chacun de tous les concepts cibles. Il appartient à la procédure d'optimisation et de réglage de paramètres des classificateurs et/ou et de fusion de sélectionner et/ou de pondérer les éléments pertinents. L'apprentissage et le réglage des paramètres est réalisé sur un ensemble de données d'apprentissage annotées. Une fois ces deux procédures réalisées, le système peut être appliqué sur de nouveaux échantillons non vus antérieurement pour leur assigner des scores de détection. Les types exacts de descripteurs, de classificateurs et de méthodes de fusions utilisés n'ont pas besoin d'être spécifiés à ce stade.

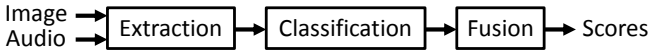


FIGURE 3.5 – L'architecture du système de référence utilisé.

Par rapport aux notations présentées dans la section 3.1, nous proposons de modéliser notre approche par un 7-tuple comme décrit dans ce qui suit :

$$< C, E(E_D \cup E_T), A, F_{descr}, F_{sc}, , F_c, F_{ac} >$$

Nous rappelons que notre but est de proposer une approche générique. C'est la raison pour laquelle nous nous focaliserons dans ce qui suit sur la définition de la fonction : F_c, puisque les autres paramètres dépendent du système utilisé. Nous parlerons aussi de certains détails concernant le module de fusion $Module_{fuse}$. Le reste des paramètres seront évoqués dans le cadre d'instanciation de notre modèle dans la partie expérimentation où nous donnerons des détails sur notre système utilisé.

Dans un système d'indexation classique, le traitement est réalisé d'une manière complètement indépendante pour tous les échantillons vidéo et pour tous les concepts cibles. La fusion est faite seulement entre les descripteurs ou les méthodes d'apprentissage, donc la fusion entre les concepts n'est pas considérée à ce stade. Une telle approche ne prend pas en compte les relations temporelles entre les échantillons vidéo (plans) et les relations sémantiques et/ou statistiques entre les concepts cibles. Comme alternative aux approches existantes, nous concevons un descripteur conceptuel qui est une version normalisée du vecteur contenant les scores de détection produits par le système de référence qu'on souhaite améliorer. Le descripteur est similaire aux "vecteurs modèles" proposés par Smith et al. [SNN03]. Après une étape de normalisation, ce descripteur de haut niveau est ajouté à l'ensemble des descripteurs de bas niveaux provenant directement des échantillons (i.e., Image ou signal audio), comme monté dans la figure 3.6.

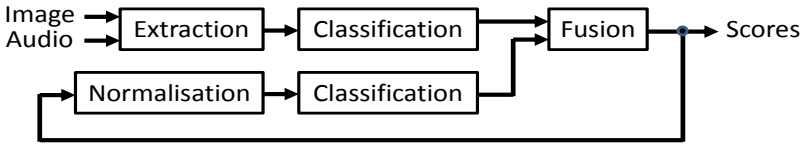


FIGURE 3.6 – Système d'indexation sémantique avec rétroaction conceptuelle.

Le descripteur de haut niveau résultant diffère des "vecteurs modèles" proposés par Smith et al. [SNN03] en deux choses : le traitement de normalisation et comment le descripteur est utilisé ensuite. Dans [SNN03], les auteurs proposent de normaliser dans un premier temps les scores de détection des concepts. Une autre normalisation est appliquée après la concaténation des différentes sorties de détecteurs, pour favoriser ou défavoriser les valeurs dominantes. D'autre part, le vecteur final est utilisé dans une étape de correspondance d'une méthode à base de distance. Les auteurs ont proposé deux mesures de distance entre la requête et les vecteurs cibles. Cependant, dans notre travail, le vecteur résultant est utilisé comme entrée d'un classificateur. Nous pensons que les méthodes d'apprentissage, surtout celles basées sur les noyaux, comme SVM (voir la section 2.7.3.2) par exemple, sont plus adaptées pour contourner le problème de normalisation des scores.

La normalisation du descripteur conceptuel peut être réalisée de différentes manières (voir la section 2.6.1). La méthode la plus simple est de centrer-normer toutes les composantes pour avoir une distribution centrée à zéro et ayant une variance égale à 1. On peut assigner aussi à chaque composante un poids, relié par exemple à la performance (par exemple AP) du système correspondant au concept concerné. Nous proposons dans notre cas d'utiliser les méthodes : "ACP" (voir la section 2.6.2.1) et "power law" (voir la section 2.6.1.1) pour normaliser les descripteurs.

Pour être utilisé comme un descripteur classique, le descripteur conceptuel doit être disponible pour le corpus de développement (apprentissage) et celui de prédiction (test). Comme il est naturellement disponible pour les données de

test, on a besoin de le générer pour les données de développement. Dans ce cas, la validation croisée est utilisée.

Le nouveau descripteur est finalement traité exactement comme ceux déjà existants (de bas niveaux) : il est utilisé pour une classification et inclus dans un processus de fusion. À ce stade, il n'y a pas de boucle dans le processus et la rétroaction est jusqu'à présent plate comme montré dans le figure 3.7. La rétroaction conceptuelle peut tout de même être appliquée plusieurs fois jusqu'à convergence. Nous détaillerons cette idée dans la section 3.6.

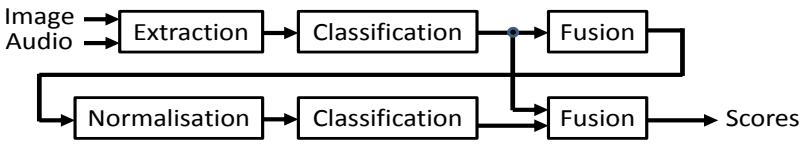


FIGURE 3.7 – Système d'indexation sémantique avec rétroaction conceptuelle, étalé (une seule itération).

Comme décrit ci-haut, la rétroaction conceptuelle ne prend pas en compte le cohérence temporelle des documents vidéos. Pour en tenir compte, nous nous basons sur la méthode du re-scoring temporel (TRS) [SQ11c, SQ11b]. TRS met à jour les scores de détection d'un concept dans un plans vidéo en prenant en compte les scores de détection du même concept dans les plans voisins. Cette méthode est présentée plus en détails dans la section 4.1 du chapitre 4. TRS est réalisée séparément et avec la même manière pour chaque concept.

Comme une perspective au système de rétroaction conceptuelle, la méthode TRS peut être vue comme une étape de post-traitement dans le module de fusion (la normalisation et l'optimisation du descripteur peuvent être vues comme une étape de pré-traitement intégrée dans le module de classification). Avec cette approche, les aspects temporel et conceptuel sont pris en compte mais séquentiellement au lieu de les considérer conjointement comme dans d'autres approches. Cela peut paraitre peu optimal mais ce n'est probablement que légèrement, parce que des expérimentations antérieures ont montré que la taille optimale de la fenêtre temporelle dépend pour un concept de l'instabilité entre les corpus de développement et de test. En plus, cette sous-optimalité serait réduite dans le cas d'une rétroaction itérative.

La nouveauté de l'approche proposée vient de l'utilisation d'un descripteur conceptuel en plus de ceux déjà disponibles (de bas niveau) et de la combinaison entre la rétroaction et le TRS de façon séquentielle de manière à capturer les aspects conceptuel et temporel conjointement.

Comme nous le montrons dans la partie relative aux expérimentations, il est judicieux de combiner les résultats de la prédiction basée sur le descripteur conceptuel avec ceux obtenus via des descripteurs de bas niveau. Nous proposons donc de fusionner ces deux résultats pour gagner en performance. Cette fusion est assurée par la fonction F_{ac} qui peut prendre différentes définitions possibles.

3.4.2 Expérimentations et résultats

Nous avons testé et évalué notre approche dans le contexte de la tâche d'indexation sémantique (voir la section 2.13.2.3) de TRECVid 2012 [OAM+12]. Le tableau 3.3 présente quelques détails concernant le corpus de données TRECVid 2012. Les annotations de 346 concepts ont été fournies pour le corpus de développement dans le cadre du travail de collaboration [AQ08b] ; et les jugements de pertinence ont été fournis pour 46 d'entre eux pour le corpus de test. La mesure d'évaluation est la précision moyenne inférée (infAP) sur les 46 concepts évalués, qui est une estimation classique de la mesure MAP obtenue en utilisant la méthode de Yilmaz et al. [YA06].

	Développement	Test
heures de vidéo	∼600	∼200
Nombre de fichiers	19,701	8,263
Nombres de plans vidéo	400,289	145,634

TABLE 3.3 – La collection TRECVid 2012 de la tâche d'indexation sémantique.

Le système de base utilisé pour l'évaluation inclue un grand nombre de descripteurs, en plus des scores de détections qui leur sont associés et qui ont été mis à disposition pour les participants TREDVid par le projet IRIM [BLS+12]. Nous avons utilisé une liste de 19 types de descripteurs suivants :

- CEALIST/tlep_576 (couleur/texture)
- CEALIST/bov_dsiftSC (sac de mots de SIFT dense pyramidal)
- CEALIST/2012_motion1000_tshot (mouvement)
- ETIS/lab (histogrammes de couleur pyramidaux)
- ETIS/qw.bin (ondelettes quaternioniques)
- ETIS/vlat_hog3s4-6-8-10_dict64
- EUR/sm462 (moments de saillance)
- INRIA/dense_sift (sac de mots de SIFT denses)
- INRIA/vlad
- LABRI/faceTracks
- LIF/percepts (percepts, catégories locales)
- LIG/hg (histogramme de couleur et une transformation de gabor)
- LIG/opp_sift (sac de mots de SIFT "opponent")
- LIG/stip (sac de mots de HOG and HOF STIP)
- LIRIS/MFCC_4096 (MFCC, audio)
- LIRIS/OCLBP_DS_4096 (descripteurs OC-LBP)
- LISTIC/SIFT_L2 (sac de mots SIFT filteré)
- LSIS/mlhmslbp_spyr (carectériqtiques multi-échelles pyramidales)
- MTPT/superpixel_color_sift_k1064 (SIFT sur des régions "superpixel")

Pour plus de détails vous pouvez vous référer à la section B.1 de l'annexe B ou à l'article [BLS+12].

Nous avons utilisé une combinaison de KNN/MSVM (voir les sections 2.7.4.2 et 2.7.3.1) classificateurs comme détecteur dans l'étape de classification. Une

étape de pré-traitements commune a été intégrée au processus de la classification ; elle inclut une normalisation combinée à une méthode réduction de dimensionnalité de type ACP [SQ13]. Nous avons utilisé une approche de fusion tardive hiérarchique [TSBB+12] comme présentée ci après. L'étape de normalisation pour le descripteur consiste tout simplement à centrer-normer les valeur, et est réalisée pour chaque composante.

Un classificateur est entrainé pour chaque couple (descripteur, concept). Sa sortie est normalisée en utilisant la méthode de Platt. La fusion tardive est réalisée hiérarchiquement par une combinaison linéaire entre les scores normalisés. Les descripteurs les plus similaires sont fusionnés en premier, et les descripteurs les moins similaires et les combinaisons antérieures sont successivement fusionnes [TSBB+12]. La fusion de descripteurs similaires est réalisée en utilisant des poids uniformes pour gagner en robustesse. La fusion des descripteurs dissimilaires est faite en utilisant comme poids, les valeurs de AP, ou des poids optimisés par validation croisée sur le corpus de développement.

Dans le système de référence considéré ici, la similarité entre les descripteurs, ainsi que la séquence des fusions hiérarchiques sont faites manuellement par rapport au type des caractéristiques représenté dans le descripteur. L'ordre considéré est comme suit : fusionner les classificateurs KNN et MSVM, fusionner les variantes du même descripteur (c-a-d. Le même descripteur de type sac de mots avec différentes tailles du dictionnaire), fusionner les descripteurs de différents types (couleur, texture, SIFT, percepts, ...), fusionner les différentes modalités (visuelle, audio, texte). Les méthodes basées sur les similarités estimées et le regroupement (clustering) peuvent être considérées ici, mais elles ne sont pas efficaces dans une approche manuelle [TSBB+12].

Le descripteur conceptuel est ensuite généré à partir des scores de la fusion tardive des descripteurs de bas niveau, et est traité de la même manière qu'un descripteur extrait directement du signal vidéo. Une fusion tardive entre les résultats de prédiction du descripteur conceptuel et ceux de la fusion des tardive des descripteurs de bas niveau est effectuée dans le dernier stage (F_{ac}). Nous avons choisi pour F_{ac} une fonction de combinaison linéaire pondérée par les valeurs de précisions moyennes calculées sur un corpus de développement. Nous pouvons utiliser dans la rétroaction et la fusion les scores de détection des descripteurs de bas niveau, mais nous choisissons de réaliser ces deux opérations en utilisant uniquement les scores de la fusion tardive des descripteurs de bas niveau, parce que ces résultats présentent une meilleure performance en termes de MAP.

Les résultats sont présentés dans ce qui suit sur l'ensemble de test uniquement, mais le réglage et l'optimisation des paramètres sont effectués par validation croisée sur le corpus de développement. Les résultats sur le corpus de test sont consistants avec ceux de l'ensemble de développement.

Résultats :

Dans la première partie des expérimentations, nous avons comparé différentes stratégies de combinaison du re-scoring temporel et la rétroaction conceptuelle.

Le tableau 3.4 montre la performance en termes de infAP de différentes variantes du système.

Méthode	infAP
Fusion des descripteurs de bas niveau “niveau signal”	0.2613
Descripteur conceptuel “niveau conceptuel”	0.2422
Fusion des niveaux conceptuel et signal	0.2756
Fusion des niveaux conceptuel et signal + TRS	0.2863
Fusion “niveau signal” + TRS	0.2691
Descripteur conceptuel (TRS avant)	0.2644
Fusion des niveaux conceptuel et signal (avec TRS)	0.2925
Fusion des niveaux conceptuel et signal (avec TRS)+ TRS	0.2981

TABLE 3.4 – Combinaison de la rétroaction conceptuelle et le re-scoring temporel.

Dans la partie haute du tableau 3.4 (quatre premières lignes), nous étudions la rétroaction conceptuelle décrite dans la section 3.4 dans le cas où elle est appliquée directement sur les sorties des classificateurs (sans TRS). TRS est appliqué ensuite sur le résultat de la fusion du système de référence et la rétroaction conceptuelle, comme montré dans la figure 3.8. La performance atteinte avec le descripteur conceptuel (0.2422) est inférieure à celle du système original de base (0.2613) duquel il est dérivé, indiquant que le classificateur est incapable de faire mieux que de répliquer la composante correspondante dans le vecteur, et est également incapable de trouver que cela est une meilleure stratégie pour maximiser la mesure infAP globale. Cependant, il capture des informations différentes et complémentaires comme cela peut être vu quand les résultats sont fusionnés avec ceux du niveau signal où on atteint une amélioration significative ((0.2756-0.2613)/0.2613 : +5.5% de gain relatif). TRS postérieur produit un nouveau gain significatif ((0.2863-0.2756)/0.2756 : +3.9%). Ce gain du re-scoring temporel postérieur est similaire au gain obtenu par TRS appliqué sur la sortie de la fusion “niveau signal” ((0.2691-0.2613)/0.2613 : +3.0%).

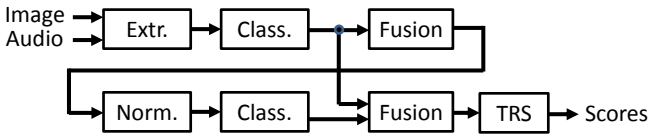


FIGURE 3.8 – Rétroaction conceptuelle avant le re-scoring temporel.

Dans la partie basse du tableau 3.4 (quatre dernières lignes), nous présentons les résultats dans le cas où la rétroaction est appliquée après le TRS comme montré dans la figure 3.9. La performance du descripteur conceptuel (0.2644) reste inférieure de celle du système de base initial duquel il est dérivé (0.2691) mais la différence n’est pas très grande, indiquant que la rétroaction conceptuelle est plus efficace après TRS. Le descripteur conceptuel donne lieu à une

nouvelle grande amélioration étant fusionné avec le système initial ((0.2925-0.2691)/0.2691 : +8.7%). L'application postérieure du TRS produit un nouveau gain mais il est moins important ((0.2981-0.2925)/0.2925 : +1.9%). Cela aurait pu être prévu du fait qu'un premier passage de TRS soit déjà inclus dans la rétroaction conceptuelle.

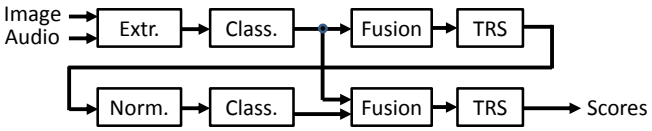


FIGURE 3.9 – Rétroaction conceptuelle après re-scoring temporel.

La performance globale du système résultant est comparable (un peu inférieure) à celle du meilleur système officiellement évalué à TRECVid 2012 sous les mêmes conditions (infAP = 0.2978) [3]. Notre proposition est très simple et facile à implémenter et, sans le re-scoring temporel qui est spécifique au cas des échantillons de type vidéo, notre approche peut être appliquée à n'importe quel système suivant le pipeline "extraction/classification/fusion" (qui est très général) pour détecter simultanément un ensemble de concepts.

3.5 Rétroaction conceptuelle étendue

La méthode de rétroaction conceptuelle décrite dans la section 3.4 peut être critiquée car elle considère un ensemble de concepts sans considérer des relations explicites avec le concept cible. D'une part, la même importance est donnée aux scores de détection de tous les concepts. En outre, tous les concepts sont pris en compte, même ceux qui n'ont aucun lien sémantique ou statistique avec le concept cible. Nous proposons dans ce qui suit deux extensions de l'approche "rétroaction conceptuelle" présentée dans la section précédente qui étudie ces deux poids abordés ci-haut. La première consiste à pondérer les dimensions conceptuelles du descripteur de haut niveau construit. La deuxième, quant à elle, consiste à filtrer les concepts et de ne prendre en compte qu'une liste spécifique contenant uniquement ceux qui sont sémantiquement ou statistiquement reliés au concept cible.

3.5.1 Approche 1 : Pondération des dimensions conceptuelles

La rétroaction conceptuelle décrite dans la section 3.4 consiste en la construction d'une descripteur conceptuel en concaténant les scores de détection d'un ensemble de concepts. Ce descripteur de haut niveau sera ensuite introduit à nouveau en entrée d'un système d'indexation et traité comme tout autre type

3. D'autres soumissions du même groupe ont atteint une valeur de $infAP = 0.3210$, mais ils ont utilisé des annotations additionnelles non officielles).

de descripteur de bas niveau. Cela génère, pour un échantillon donné, le même descripteur quelque soit le concept cible à détecter.

L'inconvénient de cette approche est le fait de donner la même importante à tous les concepts quelque soit le concept cible. Cependant, en réalité, les concepts sont connectés les uns aux autres par différents degrés. Par exemple, les concepts "roue" et "dehors" sont sémantiquement plus liés au concept "véhicule" que le concept "cuisine" et "téléphone". D'autre part, les concepts "ordinateur" et "sandwich" sont sémantique moins corrélés au concept "feu" que "explosion". Nous pensons qu'assigner un poids à chaque concept a plus de sens que de leur accorder tous la même importance. Ce poids est censé refléter le degré de relation entre le concept cible et chacun des autres concepts considérés. Nous proposons donc d'ajouter une fonction F_{weight} au modèle de la rétroaction conceptuelle classique proposé dans la section 3.4. F_{weight} permettra de pondérer le score de détection de chaque concept c par le degré de sa corrélation au concept cible :
$< C, E, A, F_{descr}, F_{sc}, F_{weight}, F_c, F_{ac} >$

F_{weight} construit donc le descripteur en concaténant les valeurs $score \times poids$ pour tous les concepts au lieu de considérer comme dans l'approche classique les valeurs $score$. Les autres paramètres restent inchangés par rapport à l'approche classique. Nous proposons d'utiliser deux sources d'information pour calculer le degré de corrélation :

1. Les annotations des exemples d'apprentissages ($corrL$)
   Seulement les annotations fournies par un humain sont utilisées pour calculer le degré de corrélation entre les concepts (voir la section 3.1 pour les notations) :

$$CorrL(c_i, c_j) = \frac{\sum_{e \in E_D} (A(e, c_i) - A(\bar{.}, c_i))(A(e, c_j) - A(\bar{.}, c_j))}{\sqrt{\sum_{e \in E_D} (A(e, c_i) - A(\bar{.}, c_i))^2} \sqrt{\sum_{e \in E_D} (A(e, c_j) - A(\bar{.}, c_j))^2}} \tag{3.9}$$

   Où $A(\bar{.}, c_k)$ représente la moyenne des valeurs $A(e, c_k)$: $A(\bar{.}, c_k) = \sum_{e \in E_D} A(e, c_k)/||E_D||$

2. Les scores initiaux de détection des concepts (F_{sc}), qui sont générés sur un corpus de développement ($CorrS$) : (voir la section 3.1 pour les notations)

$$CorrS(c_i, c_j) = \frac{\sum_{e \in E_D} (F_{sc}(e, c_i) - F_{sc}(\bar{.}, c_i))(F_{sc}(e, c_j) - F_{sc}(\bar{.}, c_j))}{\sqrt{\sum_{e \in E_D} (F_{sc}(e, c_i) - F_{sc}(\bar{.}, c_i))^2} \sqrt{\sum_{e \in E_D} (F_{sc}(e, c_j) - F_{sc}(\bar{.}, c_j))^2}} \tag{3.10}$$

   Où $F_{sc}(\bar{.}, c_k)$ représente la moyenne des valeurs $F_{sc}(e, c_k)$: $F_{sc}(\bar{.}, c_k) = \sum_{e \in E_D} F_{sc}(e, c_k)/||E_D||$

$CorrL(c_i, c_j)$ présente un inconvénient qui est l'indisponibilité des annotations pour certains échantillons non annotés. Cela donnera une matrice creuse. Dans ce

travail nous forçons l'attribution de la valeur zéro (0) pour ce genre d'exemples, même si cela fausse en quelque sorte les valeurs de corrélation.

Nous notons que dans cette approche nous ne modifions par la taille du descripteur, mais nous augmentons ou nous diminuons uniquement l'importance accordée à chaque concept, en utilisant les coefficients de corrélation comme valeurs de ces poids.

Pour résumer, nous proposons d'appliquer la rétroaction conceptuelle pour détecter un concept tc en pondérant les scores de détection de tous les concepts c_j considérés par les degrés de corrélation entre le concept cible tc et les différents concepts c_j ($score \times corrL$ ou $score \times corrS$).

3.5.2 Approche 2 : Filtrage de concepts à base de relations sémantiques

Nous présentons une deuxième extension de la rétroaction conceptuelle décrite dans la section 3.4. Au lieu de prendre tout simplement tous les concepts dans l'approche classique, ou de pondérer les dimensions conceptuelles dans l'extension décrite dans la section 3.5.1, nous proposons d'ajouter une fonction au modèle de la rétroaction classique, qui consiste à filtrer les concepts et de sélectionner une liste de concepts jugés sémantiquement liés au concept cible. Nous proposons d'utiliser une ontologie hiérarchique O prédéfinie pour les concepts visuels afin de sélectionner depuis cette hiérarchie l'ensemble des concepts ayant un lien sémantique avec le concept cible. Parmi l'ensemble des relations possibles nous retenons dans le cadre de ce travail uniquement les relations de type : "ancêtre↔fils". Un autre choix possible qui mène au même résultat sans utiliser une ontologie est de considérer la relation sémantique d'implication (e.g. $Car \Rightarrow Vehicle$), à partir du moment où "$A\ est\ fils\ de\ B$" et "B est ancêtre de A" dans l'ontologie est équivalent à "$A \Rightarrow B$". Par exemple, dans la figure 3.10 "Car est le fils de Ground_Vehicle" et "Ground_Vehicle est un ancêtre de Car" dans l'ontologie est équivalent à "Car ⇒ Ground_Vehicle".

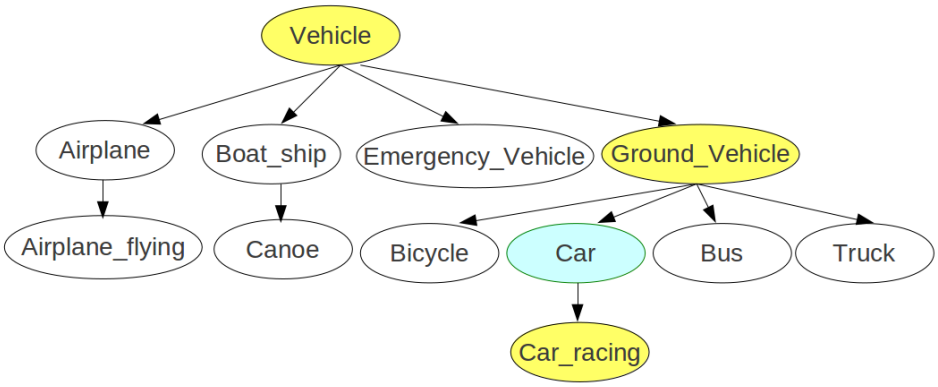


FIGURE 3.10 – Rétroaction conceptuelle étendue (2) : Sélection des ancêtres et des descendants du concept cible dans l'ontologie (e.g. Car).

Nous ajoutons une fonction F_{filter} et une ontologie au modèle de la rétroaction conceptuelle classique de la section 3.4 :
$< C, E, A, F_{descr}, F_{sc}, O, F_{filter}, F_c, F_{ac} >$

Nous proposons deux définitions à la fonction F_{filter}, pour déterminer quels sont les concepts à considérer comme sémantiquement reliés au concept cible. Ces deux approches de filtrage ont déjà été présentées dans la section 3.2.1 :

- "Ancêtres ou Descendants" : Pour un concept c_i, on considère uniquement les concepts qui sont des ancêtres ou des descendants de c_i dans la hiérarchie de l'ontologie.
  $concepts_select(c_i) = \{c_j |\ c_j$ est un ancêtre ou un descendant de c_i dans l'ontologie$\}$. La figure 3.10 présente un exemple de sélection pour le concept "Car" ;
- "Famille de concept" (voir la section 3.4). Dans la figure 3.11, la famille du concept "Car" contiendra : "Vehicle", "Emergency_Vehicle", "Ground_Vehicle", "Airplane", "Airplane_Flying", "Bicycle", "Bus", "Truck", "Car" et *Car_Racing*". De même, "Airplane_Flying" aura dans sa famille : "Airplane", "Airplane_Flying" et "Vehicle" ;

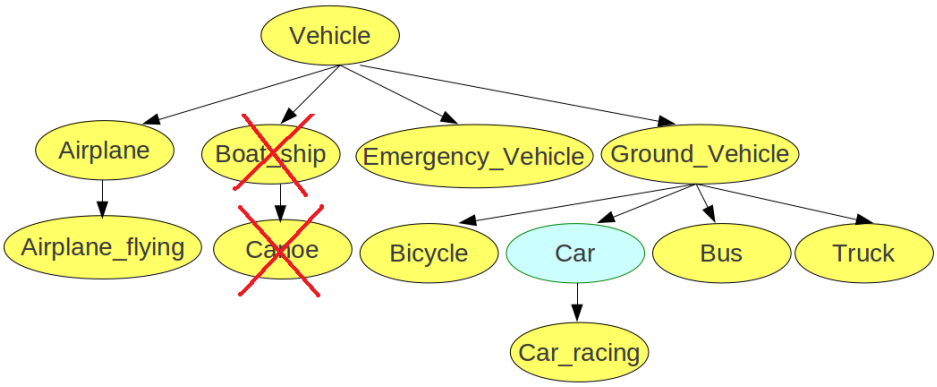


FIGURE 3.11 – Rétroaction conceptuelle étendue (2) : Sélection d'un famille du concept cible (e.g. Car).

3.5.3 Expérimentations et résultats

Pour le système initial de référence, nous avons suivi les mêmes étapes que dans l'approche classique de rétroaction conceptuelle décrites dans la section 3.4.2, dans le cas où la rétroaction est appliquée après le TRS. L'étape qui diffère concerne la construction du descripteur conceptuel. Pour cela, nous nous sommes basés sur des relations sémantiques de type "implication" pour générer la hiérarchie de concepts. Nous avons montré précédemment que ce type de relation est équivalent à la relation "Ancêtre↔fils" dans la hiérarchie d'une ontologie.

Nous avons considéré quatre cas :

1. Pondérer les scores d'un système de référence par $CorrL(c_i, c_j)$: $fb \times CorrL$;

2. Pondérer les scores d'un système de référence par $CorrS(c_i, c_j)$: $fb \times CorrS$;
3. Sélectionner les concepts qui sont des ancêtres ou des descendants du concept cible dans la hiérarchie de l'ontologie : $fb \times ancDesc$;
4. Sélectionner une famille pour chaque concept cible : $fb \times family$;

Le descripteur conceptuel résultant est introduit en entrée d'un classificateur de type MSVM (voir la section 2.7.4.2) pour générer de nouveaux détecteurs de concepts. Les scores obtenus par ces derniers sont ensuite fusionnés avec ceux du système de référence, via une fonction F_{ac} qui consiste à moyenner les scores des deux systèmes. Finalement, nous avons appliqué le re-scoring temporel (TRS) sur le résultat de cette dernière fusion pour avoir les scores finaux de détection.

Nous considérons dans ce qui suit les notations suivantes :

1. Init : système de référence, résultant de la fusion tardive des descripteurs de bas niveau ; .
2. DC : système d'indexation utilisant le descripteur conceptuel ;
3. DC+TRS : système d'indexation utilisant le descripteur conceptuel et re-scoring temporel (TRS)
4. + : Fusion des sorties des systèmes.

	DC	Init +DC	Init+"DC+ TRS"
$fb \times classique$	0.2644 (-1.7%)	0.2925 (+8.7%)	0.2981 (+10.8%)
$fb \times ancDesc$	0.1850 (-31.2%)	0.2726 (+1.3%)	0.2716 (+0.9%)
$fb \times family$	0.2508 (-6.8%)	0.2915 (+8.3%)	0.2909 (+8.1%)
$fb \times corrS$	0.2852 (+6.0%)	0.3068 (+14.0%)	0.3082 (+14.5%)
$fb \times corrL$	0.2844 (+5.7%)	0.3045 (+13.1%)	0.3045 (+13.1%)
Système initial	0.2691		

TABLE 3.5 – Résultats des approches de rétroaction conceptuelle étendue : infAP (% de gain relatif).

Le tableau 3.5 montre les résultats en termes de infAP obtenus par nos deux propositions. Nous pouvons voir que pour le "descripteur conceptuel", nous avons amélioré la performance du système initial uniquement pour $fb \times corrS$ et $fb \times corrL$ atteignant un gain relative d'environ +6% et +5.7%, respectivement. Nous remarquons aussi que, même en sélectionnant explicitement les concepts qui sont reliés au concept cible, les résultats obtenus directement de l'étape de classification sont moins bons que ceux du système initial. Dans ce contexte, $fb \times family$ est meilleure que $fb \times ancDesc$. En ce qui concerne le cas de "Fusion du système initial et du descripteur conceptuel", nous avons une amélioration dans tous les cas, mais l'approche de pondération des scores par les coefficients de corrélation est la plus efficace. À ce stade, $fb \times ancDesc$ et $fb \times family$ donnent des résultats moins bons que ceux de la rétroaction conceptuelle classique, mais $fb \times family$ fait tout de même mieux que $fb \times ancDesc$ avec un gain d'environ

+8.3%. Le meilleur résultat est obtenu en utilisant $fb \times corrS$ atteignant un gain d'environ +14.0%. Les mêmes remarques peuvent être faites pour le cas où TRS est appliqué. Encore une fois, $fb \times corrS$ a été la plus efficace réalisant un gain relatif d'environ +14.5%.

L'approche de pondération des scores de détection par les valeurs de corrélation est plus performante que la rétroaction conceptuelle classique. On peut expliquer cela par le fait que les concepts ne contribuent pas par la même quantité d'informations utiles. En effet, chaque concept a un ensemble de concepts qui stimule sa détection. Lorsque nous prenons le même poids pour tous les concepts, certains d'entre eux introduisent un bruit qui perturbe la classification et également les autres étapes.

Les approches où un filtrage basé sur le contexte sémantique a été appliqué n'ont pas donné de meilleurs résultats que ceux où tous les concepts ont été considérés. Nous pouvons donner trois hypothèses pour expliquer ce phénomène. Tout d'abord, toute information sémantique complémentaire décrivant les échantillons est utile pour les différencier. Par conséquent, plus le nombre de concepts pris en compte est grand, meilleure est la performance. Deuxièmement, dans certains cas, il n'y a pas beaucoup ou pas du tout de concepts qui sont sémantiquement liés au concept cible. Dans ce cas, les descripteurs de haut niveau sont de petites tailles ou nuls, ce qui affecte l'étape d'apprentissage. Troisièmement, il y a quelques concepts qui sont sémantiquement liés, mais qui co-occurent très rarement (voire jamais), comme par exemple "vélo" et "bateau". Dans ce cas, les relations sémantiques entre les concepts ne sont pas utiles si elles sont utilisées comme décrit dans notre approche. D'autre part, la relation de co-occurrence est importante, ce qui est confirmé par le fait que $fb \times corrS$ et $fb \times corrL$ soient plus performantes, car elles utilisent toutes les deux, la corrélation qui considère les co-occurrences de concepts.

La figure 3.12 montre les valeurs des précisions moyennes AP obtenues par nos propositions concernant la rétroaction conceptuelle (approche classique, filtrage des concepts, pondération des dimensions conceptuelles), pour certains concepts. Parce que nous ne pouvions pas, pour des raisons de clarté, présenter les détails de tous les expérimentations que nous avons faites, nous avons choisi de détailler les résultats de "Fusion du système initial et du descripteur conceptuel+TRS". Les valeurs de précisions moyennes confirment les remarques faites pendant l'analyse des MAP. En effet, nos différentes propositions améliorent un système initial qui est déjà bon en matière de performance (précision), pour l'ensemble des concepts. Les approches de pondération des dimensions conceptuelles par des coefficients de corrélation, surtout "$fb \times scXcorrL$", sont généralement les plus efficaces. Le filtrage des concepts à base de relations sémantiques améliore dans la plupart des cas le système de référence, et "$fb \times family$" fait généralement mieux que "$fb \times ancDesc$". D'autre part, on retient que les sources externes de relations sémantiques inter-concepts ne sont pas très utiles dans le cas où elles sont utilisées de la manière présentée dans ce travail. En effet, il pourrait y avoir des concepts qui ont un lien sémantique mais qui ne co-occurrent pas très souvent, comme c'est le cas de "vélo" et "bateau", qui sont tous les deux des véhicules mais qui co-occurrent rarement (voire jamais) dans plusieurs corpus. Cependant, les relations

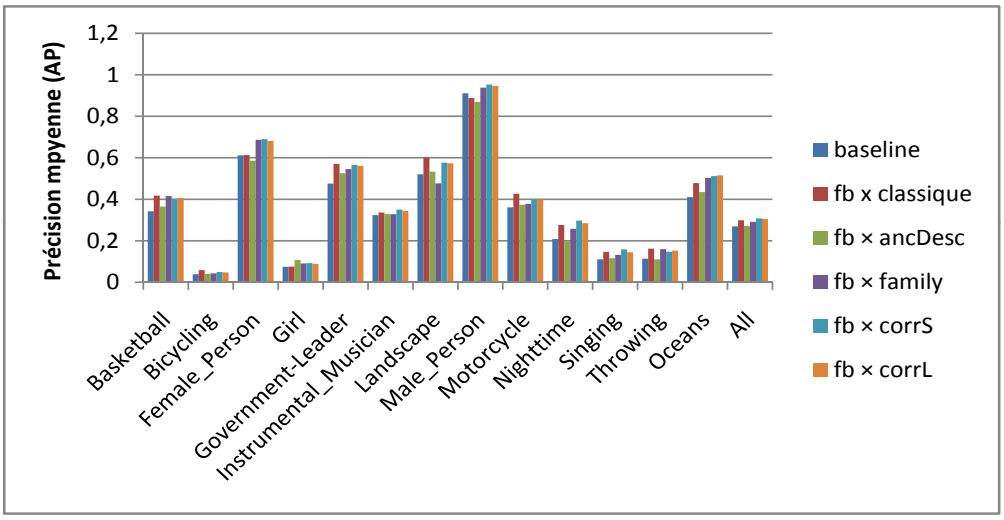


FIGURE 3.12 – Comparaison entre les résultats (précision moyenne AP) des approches de rétroaction pour certains concepts.

statistiques se révèlent très utiles et plus efficaces, parce qu'elles modélisent les relations de co-occurrences, et cela est confirmé par la supériorité des méthodes : "$fb \times scXcorrS$" et "$fb \times scXcorrL$".

Nos propositions sont très simples et faciles à implémenter et, sans le re-scoring temporel qui est spécifique au cas des échantillons de type vidéo, nos approches peuvent être appliquées à n'importe quel système suivant le pipeline "extraction/classification/fusion" (qui est très général) pour détecter simultanément un ensemble de concepts.

3.6 Rétroaction conceptuelle itérative

3.6.1 Description de l'approche

L'approche "rétroaction conceptuelle" classique ainsi que ces deux extensions peuvent être utilisées itérativement, quelque soient le concept à détecter et le type de document considérés. En effet, il suffit de tourner le processus en question en boucle au tant de fois qu'on le souhaite, comme montré dans la figure 3.13. Le résultat de l'apprenant entrainé sur des descripteurs de bas niveau est référencé par l'itération 0. Ensuite, les scores de cet apprenant sont utilisés pour construire un nouveau descripteur conceptuel à base duquel un nouveau classificateur est entrainé en utilisant le même ensemble d'annotations que dans l'itération précédente. Les résultats de cette phase sont fusionnés avec deux des résultats de l'itération 0 pour générer les résultats de l'itération 1. Le processus est ensuite itéré N fois et les résultats de chaque nouveau apprenant entrainé sur un nouveau descripteur conceptuel, sont fusionnés avec les résultats de l'itération

0 pour générer ceux de l'itération en cours. L'intuition derrière cette idée est de capturer de nouvelles informations à chaque itération. En effet, chaque nouveau apprenant entrainé sur un descripteur conceptuel est susceptible de déchiffrer de nouvelles informations sémantiques. Cependant, ces nouveaux apprenants deviendront sans doute vulnérables au bruit introduit au fur et à mesure des itérations. Dans ce contexte, nous pensons que le nombre d'itérations ne doit pas être trop grand. Le but de la fusion avec les résultats de l'itération 0 est de garder le résultat de l'apprentissage basé sur des descripteurs de bas niveau. Ces résultats sont, comme montré dans nos expérimentations antérieures, obtenus en combinant plusieurs descripteurs issus de modalités différentes. Les résultats à ce stade sont en général bons, et c'est la raison pour laquelle nous jugeons important de les garder. La fusion de ces résultats avec de nouvelles informations sémantiques apprises sur la base de descripteurs conceptuels pourrait donner de meilleurs résultats.

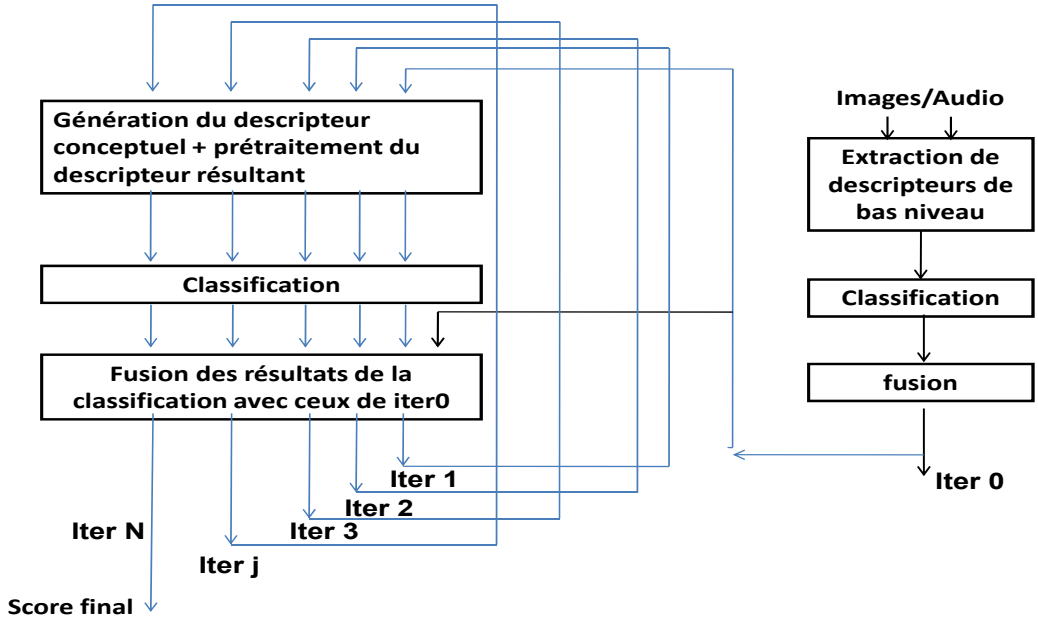


FIGURE 3.13 – Rétroaction conceptuelle itérative.

3.6.2 Expérimentations et résultats

Pour appliquer les différentes méthodes de rétroaction conceptuelle, nous avons suivi les mêmes étapes d'expérimentation décrites dans les sections : 3.4.2 pour la rétroaction classique, et la section 3.5.3 pour la version étendue. Nous avons étudié les résultats selon plusieurs niveaux d'itération. Le tableau 3.6 montre les performances du système après 0 (système de référence sans rétroaction), 1, 2 et 3 itérations, avant et après TRS. Les scores utilisés pour générer le descripteur conceptuel sont toujours pris après TRS de l'itération précédente.

	rétroaction classique		$scXcorrS$		$scXcorrL$	
	par plan	+TRS	par plan	+TRS	par plan	+TRS
iter0	0.2613 -	0.2691 +3.0%	- -	- -	- -	- -
iter 1	0.2925 +11.9%	0.2981 +14.1%	0.3068 +17.4%	0.3082 +18.0%	0.3045 +16.5%	0.3045 +16.5%
iter 2	0.2984 +14.2%	0.3014 +15.3%	0.2855 +9.3%	0.2862 +9.5%	0.2900 +11%	0.2886 +10.4%
iter 3	0.2980 +14%	0.3011 +15.2%	0.2632 +0.7%	0.2674 +2.3%	0.2656 +1.6%	0.2688 +2.9%

TABLE 3.6 – Résultats pour l'approche rétroaction conceptuelle itérative avec re-scoring temporel (TRS) : MAP et % de gain relatif. iter0 correspond au système initial de base sans appliquer la rétroaction conceptuelle.

Nous pouvons remarquer que pour la rétroaction conceptuelle classique, il y a un gain de performance uniquement dans les deux premières itérations. Le gain obtenu dans la deuxième itération est beaucoup plus faible que celui atteint dans la première, et une légère dégradation est signalée dans la troisième. Cela est probablement du au fait que la plupart des informations concernant les relations inter-concepts sont extraites dans la première itération, le reste est extrait dans la seconde, et que du bruit est amplifié dans les suivantes. Le gain apporté par TRS diminue également au fur des itérations. Ces résultats sont cohérents avec ceux obtenus par validation croisée au sein de l'ensemble de développement. Pour l'approche de pondération par les valeurs de corrélation inter-concepts, nous pouvons voir que la meilleure performance est atteinte lors de la première itération. Les résultats à ce stade sont meilleurs que ceux de la rétroaction classique. À partir de la seconde itération, les performances se dégradent de façon significative à cause du bruit accumulé. La différenciation entre les concepts par des poids de corrélation permet de capturer des informations relatives aux relations inter-concepts en une seule étape.

La rétroaction conceptuelle itérative (classique) combinée avec TRS donne un gain relatif total d'environ 15,3% (amélioration de la MAP de 0.2613 à 0.3014). Des résultats ont montré que le TRS donne un gain supérieur sur les systèmes de faibles performances que celui atteint sur des systèmes performants (voir la figure 4.4 concernant l'approche du re-scoring temporel présentée dans le chapitre 4) : plus le système initial est bon, plus il devient difficile d'améliorer ses performances. Bien qu'il n'ait pas été vérifié directement, un effet similaire concerne la rétroaction conceptuelle, et nous rappelons que dans ce travail, nous avons un système de référence qui a déjà une bonne performance.

L'approche itérative n'est pas bénéfique pour les autres variantes de la rétroaction conceptuelle. En effet, "$fb \times scXcorrS$" et "$fb \times scXcorrL$" atteignent

leurs performances maximales dans la première itération, avec un gain relatif en MAP d'environ +18.0% et +16.5%, respectivement.

La performance globale en termes de infAP du système résultant de la tâche d'indexation sémantique TRECVID 2012 est de 0,3014. À titre comparatif, le meilleur système officiellement évalué dans les même conditions a eu une valeur de infAP égale à 0,2978 [4]. Bien que les résultats rapportées dans ce travail n'ont pas été officiellement soumis à TRECVID, ils ont été réalisés exactement dans les mêmes conditions et en utilisant le même protocole expérimental et les mêmes outils d'évaluation.

3.7 Conclusion

Nous avons présenté dans ce chapitre nos contributions pour l'utilisation du contexte sémantique pour l'indexation des documents multimédia. La première approche intitulée "re-ordonnancement sémantique", reclasse les résultats d'une première classification en modifiant le score de détection calculé dans la première étape, en se basant sur des relations inter-concepts qui peuvent être issues d'une hiérarchie d'une ontologie ou d'une liste de relations de type "c_1 implique c_2". La mise à jour des scores se fait via ne fonction simple (combinaison linéaire) en impliquant des coefficients de corrélations inter-concepts calculés à partir d'un corpus de développement. Même si cette méthode a amélioré un système ayant déjà une bonne performance, le gain n'était pas spectaculaire.

La deuxième contribution nommée "reclassement sémantique par regroupement", modélise le contexte sémantique en regroupant les exemples selon leurs contenus sémantiques issus d'une étape préalable d'apprentissage. Elle se base sur l'hypothèse que si on décrit les individus par leurs contenus sémantiques, les échantillons ayant un lien sémantique se verront regroupés dans l'espace. La mise à jour des scores se fait via une combinaison de valeurs calculées après une étape de regroupement (clustering) basé sur une description des données par les scores de détection d'un ensemble de concepts. Même si cette méthode a pu améliorer un bon système d'indexation, elle contient plusieurs paramètres et reste sujette au problème du sur-apprentissage.

Notre troisième contribution appelée "rétroaction conceptuelle" consiste à construire un nouveau descripteur de haut niveau à partir des scores obtenus via une première étape de classification. Cette approche met à jour les scores de détection du concept cible en considérant une nouvelle étape d'apprentissage qui utilise le descripteur conceptuel généré. Nous avons proposé ensuite, une extension de cette approche, que nous avons appelée "rétroaction conceptuelle étendue". Cette dernière suit les mêmes étapes que la rétroaction conceptuelle classique, mais diffère sur la méthode de construction du descripteur conceptuel. Nous avons élaboré deux stratégies. Dans la première, nous avons effectué une sélection explicite des concepts à prendre en compte en ne gardant que ceux qui ont un lien sémantique avec le concept cible. Nous nous sommes basés pour effectuer cette

4. D'autres soumissions du même groupe ont atteint un infAP=0,3210, mais en utilisant des annotations supplémentaires non officielles.

sélection sur la hiérarchie d'une ontologie. Dans la deuxième stratégie, nous avons pondéré les dimensions conceptuelles, en donnant une importance aux concepts qui est relative à leurs corrélations par rapport au concept cible. Toutes les versions proposées de la rétroaction conceptuelle ont donné une amélioration significative du système de référence, mais la stratégie de pondération des dimensions conceptuelles s'est avérée plus efficace et a donné les meilleurs résultats. Finalement, nous avons montré que tous les approches de rétroaction présentées peuvent être lancées d'une manière itérative. Une étude des résultats des itérations a montré que les résultats de la rétroaction classique sont améliorés au fur des itérations, mais la performance a tendance à chuter à partir de la deuxième itération, suite à une saturation due au cumul du bruit. La rétroaction étendue ne tire pas profit de l'approche itérative, et la stratégie de pondération des dimensions conceptuelles arrive à des résultats comparables à ceux de la deuxième itération de la version classique, en une seule itération.

Toutes nos approches sont applicables sur n'importe quel système d'indexation d'images ou de vidéos par détection de concepts. Même si nos approches ont été testées sur des corpus vidéos, elles restent toute de même applicables sur des corpus d'images, sans aucune modification dans l'aspect global présenté, à part le fait qu'il faudrait omettre le re-scoring temporel qui est restreint aux vidéos ou aux documents dotés d'un aspect temporel.

Chapitre 4

Contributions pour l'utilisation du contexte temporel

Nous présentons dans ce chapitre nos contributions pour l'utilisation du contexte temporel pour l'indexation sémantique des vidéos. Le contexte temporel est défini dans le cadre de documents non fixes (e.g. Vidéos, audio, etc.). L'aspect temporel de ces documents donne la possibilité de les segmenter en échantillons dans l'axe du temps ; on peut prendre l'exemple de la segmentation de vidéos en plans. Ces échantillons sont sémantiquement liés les uns aux autres, surtout ceux qui sont temporellement proches. Le contexte temporel considère l'homogénéité des parties successives dans ce genre de documents.

Nous avons détaillé l'utilisation du contexte pour l'indexation des vidéos dans la section 2.11.3 du chapitre 2. Nous avons vu que le contexte temporel peut être exploité dans différents niveaux d'un système d'indexation, comme le montre la figure 4.1. Dans ce chapitre, nous présentons deux méthodes exploitant le contexte temporel. Dans la première, nous reprenons la méthode de re-scoring temporel décrite dans [SQ11c, SQ11b], qui exploite l'homogénéité inter-plans vidéo pour la mise à jour de scores de détection de concepts dans les plans vidéo. Cette méthode s'inscrit dans la catégorie des approches de ré-ordonnancement qui agit dans un niveau de post-classification (TRS dans la figure 4.1). Notre deuxième contribution s'inscrit parmi les approches exploitant le contexte temporel lors de la création des descripteurs (ETC dans la figure 4.1) que nous appellerons par la suite : "descripteurs temporels". Nous établissons ensuite une comparaison entre les deux méthodes "re-scoring temporel" et "descripteurs temporels". Nous comparons ensuite ces deux propositions.

4.1 Re-scoring temporel

4.1.1 Description de l'approche

En plus de l'audio, une vidéo a une caractéristique qui la rend différente d'une image fixe : l'aspect temporel. Ignorer cette caractéristique importante dans la détection de concept fait perdre des informations pertinentes. En effet, contrairement à une image les plans d'une vidéo sont liés. Safadi et al. [SQ11c,

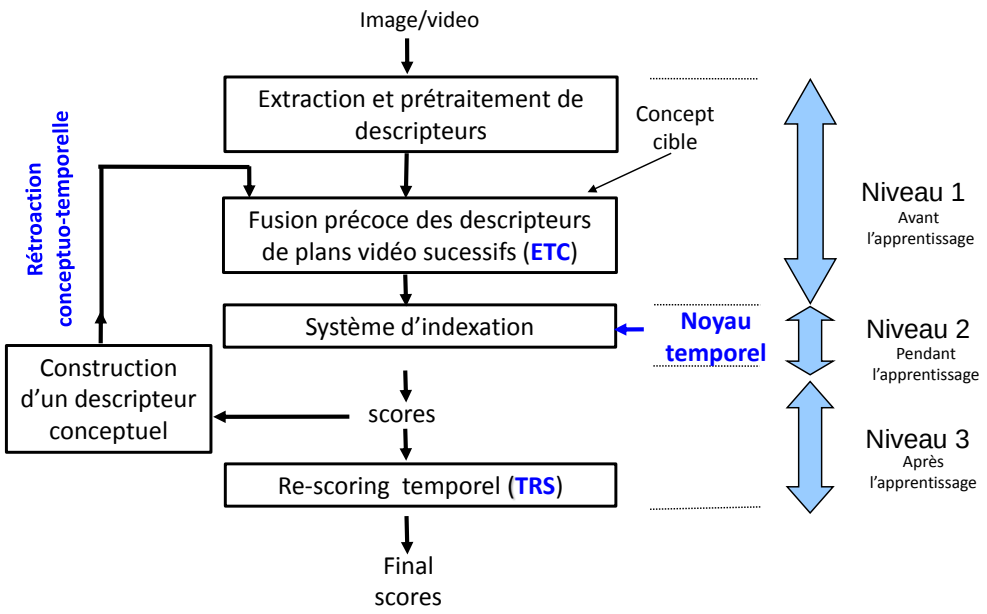


FIGURE 4.1 – Différents niveaux d'exploitation du contexte temporel.

SQ11b] ont utilisé la notion du contexte temporel en exploitant les scores de détection de concepts dans les plans voisins, arrivant à une amélioration très significative. Ce bon résultat peut être expliqué par la dépendance des contenus entre les plans successifs localement homogènes. La figure 4.2 décrit cette idée.

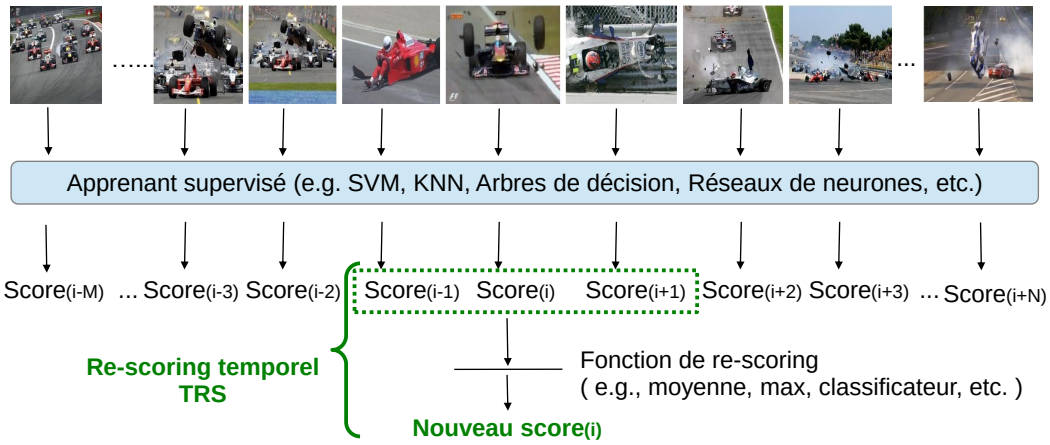


FIGURE 4.2 – Re-scorig temporel pour l'indexation sémantique des vidéos.

Nous proposons d'utiliser cette idée et d'étendre la recherche d'un concept sur une fenêtre de taille $2 \times w + 1$ (w plans précédant et w plans suivant le plan courant). Le nouveau score de détection d'un concept c_i (F_{sc}) dans le plan e est donné par la formule simplifiée suivante, tout en se basant sur les notations

décrites dans la section 3.1 :

$$F_{ac}^{F_v(e),F_p(e)}(e,c_i) = F_{sc}^{F_v(e),F_p(e)}(e,c_i) + F_{c}^{F_v(e),F_p(e)}(e,c_i) \tag{4.1}$$

avec :

$$F_{c}^{F_v(e),F_p(e)}(e,c_i) = \sum_{k=-w,k\neq 0}^{w} F_{sc}^{F_v(e),F_p(e)+k}(e,c_i) \tag{4.2}$$

Cette fonction de fusion des scores des plans vidéo voisins n'est pas unique. En effet, plusieurs fonctions sont possibles, comme celle utilisée dans [SQ11c, SQ11b], où le nouveau score de détection d'un concept c_i dans un plan e est calculé de la manière suivante :

$$F_{ac}^{F_v(e),F_p(e)}(e,c_i) = [F_{sc}^{F_v(e),F_p(e)}(e,c_i)]^{1-\gamma} \times [Z_{F_v(e),F_p(e)}]^{\gamma} \tag{4.3}$$

où γ est un paramètre qui commande la robustesse de la méthode de reclassement, et $Z_{F_v(e),F_p(e)}$ est un score global calculé à partir des scores des plans voisins (F_sc) via une fonction simple telle que la moyenne arithmétique, la moyenne géométrique, min, max, etc. γ est réglé par validation croisée sur la collection de développement.

Des expérimentations ont montré que la fonction 4.3 donne de meilleurs résultats que la première, mais la différence n'est pas significative.

4.1.2 Expérimentations et résultats

Parce qu'il s'est avéré très efficace, nous avons incorporé le re-scoring temporel dans la plupart de nos expérimentations décrites dans le travail de cette thèse. Nous n'allons donc pas décrire un protocole expérimental fixe, mais nous allons donner uniquement le schémas général suivi. La seule différence en termes d'aspect expérimental entre les résultats que nous allons présentés par la suite se situe dans le choix et le nombre de descripteurs, le nombre de résultats fusionnés pour avoir le résultat final, et la collection de données utilisée. Le système de référence utilisé et que nous voulons améliorer est décrit dans la figure 4.3.

Nous avons utilisé dans les différentes expérimentations plusieurs types de descripteurs issus de différentes modalités : visuelle, audio, mouvement, comme par exemple des descripteurs de couleurs (e.g. Histogramme RGB), de texture (e.g. Transformation de Gabor), descripteurs de l'audiio (e.g. Profil spectral), SIFT (e.g. Opponent sift), etc. En ce qui concerne les méthodes d'apprentissage, nous avons privilégié les méthodes $K-$ plus proches voisins (KNN) et MSVM pour leurs bons résultats dans le cadre de la détection de concepts visuels dans les images/vidéos [YH08a, SQ10].

Notre évaluation a été menée sur les collections TRECVid 2010 et/ou TRECVid 2012. Nous avons déroulé nos expérimentations sur un corpus de développement divisé en deux parties : Une pour l'apprentissage et l'autre pour l'évaluation ("validation croisée à 1 pli", "1-fold cross-validation" en anglais). Les annotations ont été fournies par l'annotation collaborative organisée par les équipes LIG et LIF [AQ08a]. Nous avons utilisé un lexique contenant 130 concepts pour la collection 2010 et 256 concepts pour la collection 2012.

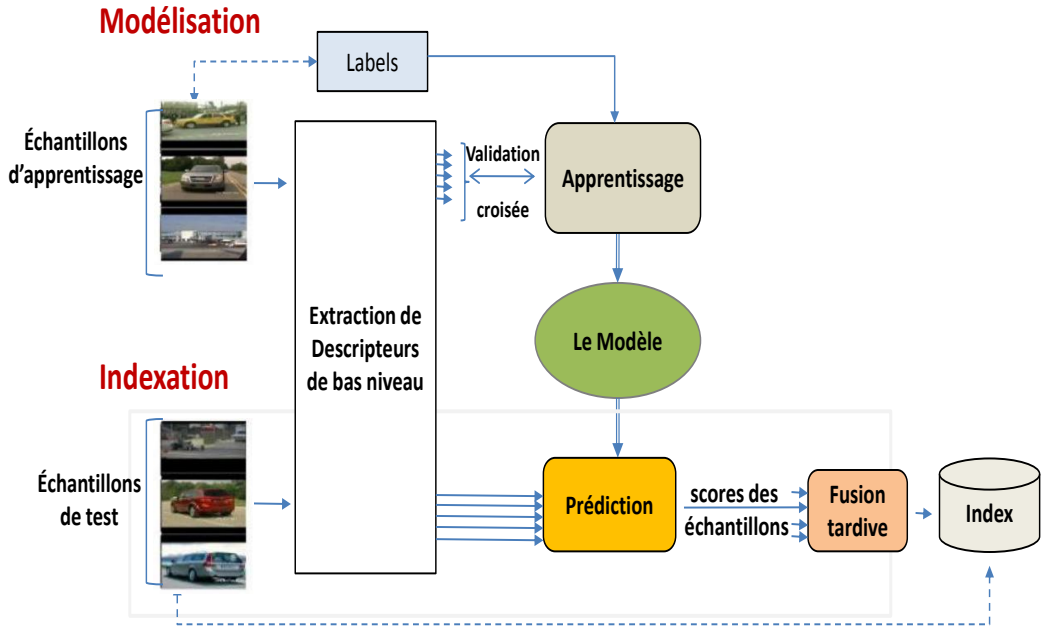


FIGURE 4.3 – Description générale du système d'indexation utilisé.

Résultats :

Le tableau 4.1 présente les résultats obtenus par la méthode du re-scoring temporel appliquée sur plusieurs systèmes et deux collections TRECVid différentes : 2010 et 2012.

Nous remarquons que le re-scoring temporel améliore les résultats quelque soit la performance du système initial. Le pourcentage du gain relatif sur le MAP varie entre +2% et +16%. La performance de l'approche étudiée peut dépendre de celle des systèmes initiaux qu'on veut améliorer. En effet, nous pouvons voir dans le tableau 4.1 que le gain relatif est inversement proportionnel à la performance du système initial. La figure 4.4 qui décrit la variation du pourcentage du gain relatif par rapport aux valeurs de la précision moyenne (MAP) des systèmes initiaux étudiés, confirme cette remarque. En effet, plus la performance du système à améliorer est bonne, plus le gain en performance obtenu par le re-scoring temporel est faible. Cela peut être expliqué par le fait qu'il est plus facile d'améliorer un système qui est mauvais/moyen plutôt qu'un système ayant une bonne performance. D'une autre, nous rappelons que la MAP n'est pas calculée sur le même nombre de concepts pour les collections 2010 et 2012 (130 concepts pour la collection TRECVid 2010 et 46 concepts pour celle de 2012), cela peut expliquer la différence entre les performances, mais pour le confirmer une étude détaillée des valeurs de AP est nécessaire. Nous tenons à préciser que le gain obtenu par le re-scoring temporel varie d'un concept à un autre, mais cette approche améliore dans la plus part des cas n'importe quel détecteur de concept dans les vidéos.

Le ré-ordonnancement temporel améliore davantage les résultats que le ré-ordonnancement sémantique présenté dans la section 3.2 du chapitre 3. Nous pouvons expliquer la différence des performances des deux approches par le fait

Système	Collection	MAP du système initial	MAP après le re-scoring temporel
Système 1	TRECVid 2010	0.0343	0.0398 **(+16.03)**
Système 2	TRECVid 2010	0.0307	0.0337 **(+9.77)**
Système 3	TRECVid 2010	0.0548	0.0608**(+10.95)**
Système 4	TRECVid 2010	0.0698	0.0782 **(+12.03)**
Système 5	TRECVid 2010	0.0136	0.0157 **(+15.44)**
Système 6	TRECVid 2010	0.0832	0.0925 **(+11.18)**
Système 7	TRECVid 2010	0.1428	**0.1561(+9.31)**
Système 8	TRECVid 2012	0.2613	**0.2691(+2.98)**
Système 9	TRECVid 2012	0.2925	**0.2981(+1.91)**

TABLE 4.1 – Résultats (MAP et % du gain relatif) du re-scoring temporel.

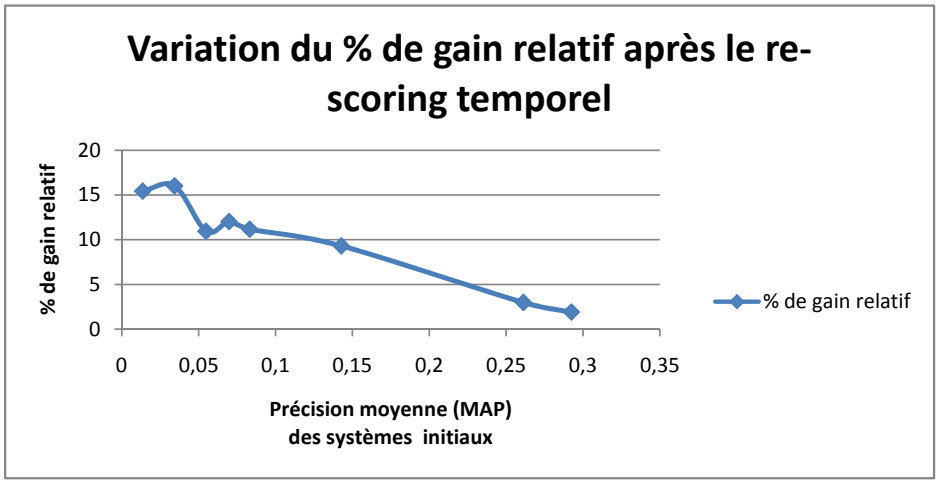


FIGURE 4.4 – Variation du % du gain relatif obtenu par le re-scoring temporel.

que le re-scoring temporel fusionne des scores obtenus par un même classificateur, mais la méthode “famille de concept” quant à elle, fusionne des scores obtenus par des classificateurs différents entrainés indépendamment et se heurte au problème de normalisation des scores.

4.2 Descripteurs temporels

Nous avons vu dans la section 2.11.3 du chapitre 2 et la section 4.1 de ce chapitre 4, que le contexte temporel s’avère très utile pour l’indexation sémantique des vidéos. Nous proposons dans cette section une autre approche qui consiste à incorporer de l’information temporelle de bas niveau dans le but de améliorer la performance d’un système d’indexation de vidéos.

4.2.1 Description de l'approche

L'approche du re-scoring temporel que nous avons présentée dans la section 4.1 utilise de l'information sémantique, qui est un ensemble de scores de détection d'un concept visuel dans les plans voisins. La performance de cette approche est bonne dans le cas où les détecteurs de concepts dans les plans ont une performance raisonnablement bonne. Cependant si on prend des détecteurs de concepts avec des performances médiocres on peut être surpris par une dégradation de performance. Autrement dit, si par exemple un détecteur prévoit par erreur l'occurrence d'un concept dans les plans s_{i-1} et s_{i+1}, il peut améliorer la probabilité que le plan s_i contienne le concept cible alors que ce dernier n'appartient pas au plan s_i. Cette erreur de propagation temporelle de l'information sémantique est dûe aux erreurs commises par les détecteurs du concept cible dans les plans dans une étape antérieure de classification. D'autre part, nous avons également vu précédemment que le gain atteint par le re-scoring temporel est inversement proportionnel à la performance du système de référence. De plus, cette méthode s'avère sujette au cumul du bruit. Nous proposons dans cette section une méthode utilisant le contexte temporel dans une étape qui échappe au problème de la propagation des erreurs de classification.

Comme montré dans la figure 4.1, il est possible d'incorporer le contexte temporel dans l'étape d'extraction des descripteurs dans le but d'avoir des descripteurs qui considèrent eux même l'aspect temporel. Les descripteurs de mouvements en sont un exemple typique. Une manière de le faire est de considérer une fenêtre temporelle lors de l'extraction du descripteur au lieu de ne prendre en compte que la partie relative au plan en question. Par exemple, pour calculer un histogramme de couleur, on peut compter les occurrences des valeurs des niveaux de gris non seulement dans le plan (ou image(s) clé(s)) en question, mais en étendant le compte dans les plans voisins également. Pareillement, on peut étendre pour le calcul des SIFT, l'extraction des points d'intérêts dans les segments temporels voisins et aussi le compte des occurrences des mots visuels lors du calcul des sacs de mots. Bien qu'avec une telle idée on code de l'information temporelle, la notion de l'ordre est omise. En effet, après le calcul de l'histogramme sur une fenêtre temporelle, on est incapable de différencier l'information relative au plan courant de celles correspondant aux plans voisins. Il devient donc impossible de pondérer les informations issues de différentes sources. Pour remédier à ce problème nous proposons une approche qui échappe à l'inconvénient de la propagation des erreurs de classification, et est beaucoup plus simple à mettre en œuvre.

Nous proposons d'extraire les descripteurs d'une manière classique, c'est-à-dire, de chaque segment (plan, image, segment temporel, etc) et de réaliser ensuite une fusion précoce (voir la section 2.8.1), en concaténant pour chaque plan, les descripteurs des plans voisins appartenant à une fenêtre temporelle de taille égale à $(2 \times w + 1)$ plans consécutifs et centrée sur le plan concerné (w plans précédents et w plans suivants). La figure 4.5 décrit le processus pour une fenêtre temporelle de taille égale à trois plans successifs.

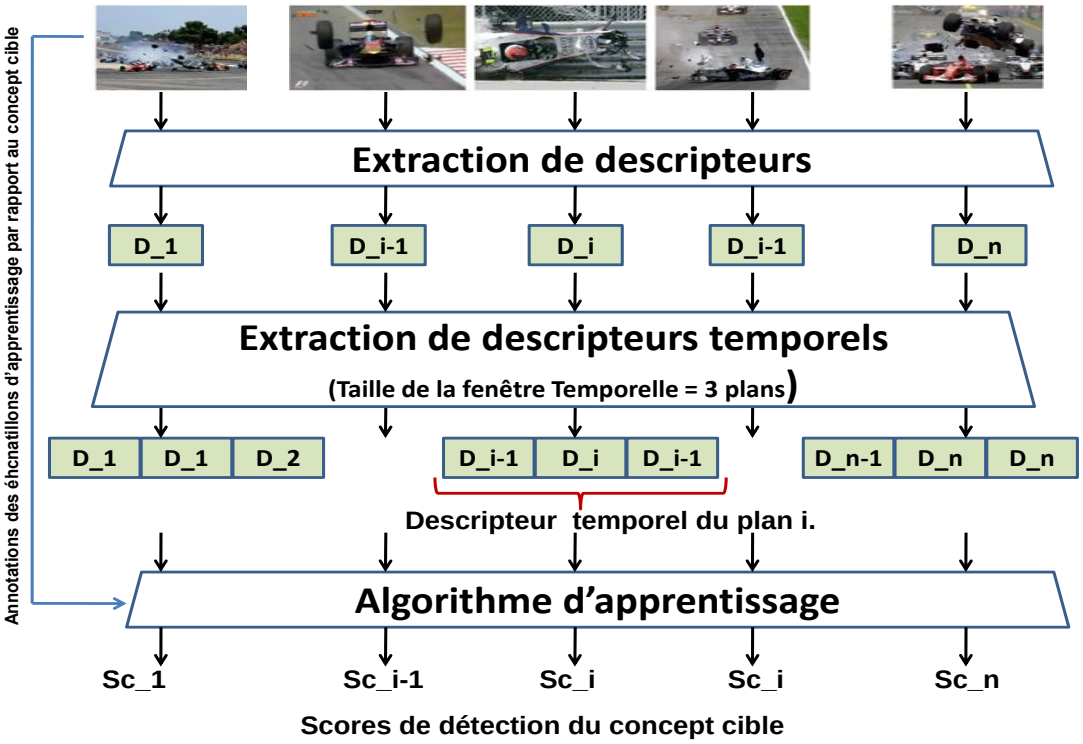


FIGURE 4.5 – Principe de construction des descripteurs temporels. La fenêtre temporelle est centrée sur le plan courant et sa taille est égale à trois plans consécutifs.

Bien qu'elle soit simple, cette approche se heurte à un grand problème, qui est la taille du descripteur résultant. En effet, le descripteur final est $(2 * w + 1)$ fois plus grand que le descripteur classique extrait de chaque plan dans l'étape antérieure. Ce point peut devenir critique surtout pour certains types de descripteurs de grande taille comme, par exemple, les "bags of opponent-SIFT" qui peuvent avoir une taille d'environ 2000 composantes. Ainsi pour une fenêtre temporelle de taille 11 (w=5, donc 5 plans précédents et 5 plans suivants), on peut se retrouver avec un descripteur de 22 000 composantes. Cela affecte l'étape d'apprentissage qui dépend de la taille des descripteurs et du nombre d'échantillons positifs et négatifs et aussi de la stratégie de l'algorithme d'apprentissage. Il est donc recommandé de prévoir une étape de réduction de dimensionnalités (voir la section 2.6.2 du chapitre 2). D'autre part, il n'est pas recommandé de considérer une grande fenêtre temporelle pour les mêmes raisons de performance de calcul.

Nous proposons dans un second temps de pondérer les informations selon leur ordre chronologique, de façon à donner plus d'importance à l'information issue du plan courant, et de diminuer le poids des informations contextuelles des plans voisins. Plus le plan est loin du centre de la fenêtre temporelle, plus le poids qui lui est attribué est faible. La figure 4.6 décrit ce processus de pondération. Les valeurs des composantes de chaque fragment temporel du descripteur temporel sont pondérées par la valeur du poids qui lui est associée.

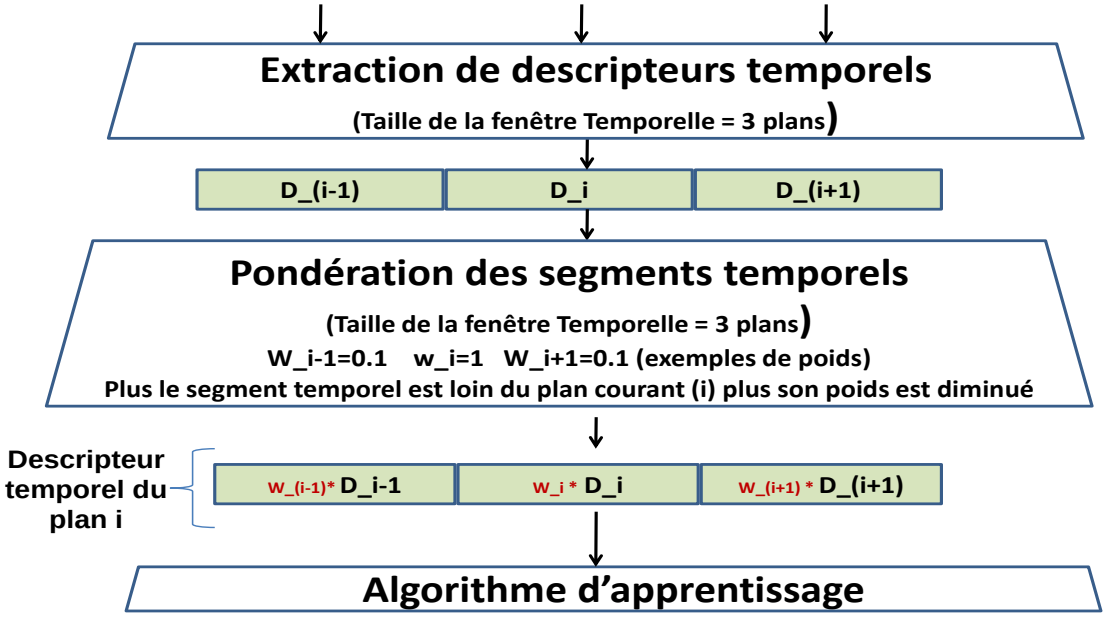


FIGURE 4.6 – Pondération des segments temporels du descripteur temporel du i ème plan. La fenêtre temporelle est centrée sur le plan courant et sa taille est égale à trois plans consécutifs.

4.2.2 Expérimentation et résultats

Nous avons testé et évalué notre approche proposée dans le contexte de la tâche d'indexation sémantique (voir la section 2.13.2.3) de TRECVid 2012 [OAM+12]. Les annotations de 346 concepts ont été fournies pour le corpus de développement dans le cadre du travail de collaboration [AQ08b] ; et les jugements de pertinence ont été fournis pour 46 d'entre eux pour le corpus de test. Le mesure d'évaluation est la précision moyenne inférée (infAP) sur les 46 concepts évalués, qui est une estimation classique de la mesure MAP obtenue en utilisant la méthode de Yilmaz et al. [YA06].

Nous avons utilisé une liste de 10 variantes des cinq types de descripteurs suivants :

- LIG/hg104 (hg) : fusion précoce (concaténation) d'un histogramme couleurs : LIG/$h3d$64 et d'une transformation de Gabor : LIG/gab40 (104-dim) ;
- LIG/opp_sift (sac de mots de SIFT "opponent")
- INRIA/dense_sift (sac de mots de SIFT denses)
- INRIA/vlad
- LISTIC/SIFT_L2_retinaMasking : sac de mots basé sur des SIFT en utilisant un modèle de rétine en temps réel.

Pour plus de détails sur ces descripteurs vous pouvez vous référer à la section B.1 de l'annexe B ou à l'article [BLS+12] pour une description plus détaillée.

Nous avons choisi d'utiliser MSVM comme classificateur/détecteur dans l'étape de classification. Une étape de pré-traitements commune a été intégrée au pro-

cessus de la classification ; elle inclut une normalisation combinée à une méthode de réduction de dimensionnalité de type ACP [SQ13].

Après avoir extrait les dix descripteurs de chaque plan vidéo, nous avons entrainé MSVM pour générer un modèle de référence. Les résultats des dix descripteurs pour chaque couple (concept, plan) sont ensuite fusionnés en moyennant les différents scores obtenus. Nous appellerons par la suite ce résultat “Descripteurs classiques sans TRS”.

Nous avons ensuite généré les descripteurs temporels comme décrit dans la section 4.2.1. Parce qu'il a été difficile de réaliser les expérimentations avec une fenêtre temporelle de grande taille, nous avons considéré pour chaque plan uniquement le plan qui le précède et celui qui le suit (fenêtre temporelle de taille égale à trois plan consécutifs). C'est pour cette raison que nous n'avons retenu que les dix descripteurs cités précédemment. Nous avons fixé les choix des poids après des essais sur des données de développement sur un descripteurs de taille initiale réduite. Nous avons considéré par la suite, trois cas :

- $w_{i-1} = 1$, $w_i = 1$ et w_{i+1}=1
- $w_{i-1} = 0.05$, $w_i = 1$ et w_{i+1}=0.05
- $w_{i-1} = 0.01$, $w_i = 1$ et w_{i+1}=0.01

Nous avons généré pour chacun des dix descripteurs un descripteur temporel. Donc, pour chacun de ces trois cas, nous avons obtenu dix descripteurs temporels.

Nous avons entrainé ensuite de nouveaux détecteurs de concepts de type MSVM (voir la section 2.7.4.2) sur les descripteurs temporels. Pour chacun des trois cas et chaque couple (concept, plan), nous avons fusionné les résultats obtenus par les dix descripteurs temporels, en moyennant les scores issus de l'étape de prédiction. Nous appellerons par la suite ce résultat “Descripteurs temporels sans TRS”.

Nous avons considéré comme système de référence, la fusion des résultats obtenus par les dix descripteurs classiques (sans aspect temporel).

Nous comparons l'approche proposée avec la méthode du re-scoring temporel présentée dans la section 4.1, pour étudier la complémentarité de différentes approches exploitant le contexte temporel. Nous avons donc appliqué un re-scoring temporel (TRS) sur les résultats “Descripteurs classiques sans TRS” et “Descripteurs temporels sans TRS”, respectivement ; pour générer de nouveaux résultats que nous appellerons par la suite “Descripteurs classiques + TRS” et “Descripteurs temporels +TRS”, respectivement.

Résultats

Le tableau 4.2 présente les résultats obtenus, en termes de précision moyenne calculée sur les 46 concepts étudiés. La première colonne des valeurs de MAP présente les résultats obtenus par la fusion des sorties des classificateurs entrainés sur les descripteurs temporels sans application du TRS. La deuxième colonne des valeurs de MAP, quant à elle, décrit les résultats de l'application du re-scoring temporel sur les résultats de la fusion des dix différents descrip-

teurs classiques/temporels. La dernière ligne du tableau concerne la fusion des descripteurs classiques (sans aspect temporel).

			Fusion sans TRS	Fusion + TRS
Systèmes			MAP	
Descripteurs temporels	Poids	Plan précédent = 0.01 Plan courant = 1 Plan suivant = 0.01	0.2330	0.2365
		Plan précédent = 0.05 Plan courant = 1 Plan suivant = 0.05	0.2258	0.2285
		Plan précédent = 1 Plan courant = 1 Plan suivant = 1	0.2052	0.2074
Descripteurs classiques			0.2312	0.2375

Table 4.2 – Résultats (MAP) des approches du re-scoring temporel et des descripteurs temporels sur la collection TRECVid 2012.

La première remarque qu'on peut faire est que l'étape de pondération affecte considérablement les résultats. En effet, nous pouvons voir dans le tableau 4.2 qu'en attribuant le même poids (=1) aux différents segments temporels on obtient un résultat moins bon qu'une simple utilisation des descripteurs classiques. L'écart entre ces deux résultats est important atteignant une différence relative d'environ -11.24%. En diminuant les poids des plans se trouvant à droite et à gauche du plan courant, la performance du système s'améliore jusqu'à dépasser la performance du système fusionnant les descripteurs classiques. Le gain relatif à ce stade est d'environ 0.77%. Nous rappelons que dans les résultats présentés dans le tableau 4.2, nous avons considéré uniquement le plan précédant et celui qui suit le plan courant. D'autre part, le TRS améliore le résultat dans tous les cas, comme nous l'avons montré dans la section 4.1.2. Le gain relatif varie entre+1% et +2.72%.

Deux autres remarques importantes peuvent être faites en analysant ces résultats. La première est le fait que le re-scoring temporel améliore davantage le résultat de la fusion des descripteurs classique que l'utilisation des descripteurs temporels, en atteignant un gain relatif d'environ +2.72% ((0.2375-0.2312)/0.2312) et 0.77% ((0.2330-0.2312)/0.2312), respectivement. La deuxième remarque est que les résultats des descripteurs temporels bénéficient d'une amélioration suite à l'application du TRS dans un post-traitement, atteignant un gain relatif supplémentaire d'environ +1.5% ((0.2365-0.2330)/0.2330).

4.3 Conclusion

Nous avons présenté dans ce chapitre deux méthodes exploitant le contexte temporel pour l'indexation des vidéos par détection de concepts. La première ap-

proche est une méthode de ré-ordonnancement qui consiste à modifier les sorties d'un détecteur d'un concept en exploitant les scores de sa détection dans les plans voisins, en les combinant au score relatif au plan en question, pour générer un nouveau score de détection du concept cible. Cette méthode qui utilise des informations sémantiques a montré son efficacité en améliorant plusieurs systèmes de référence en atteignant des gains relatifs inversement proportionnels à la performance des systèmes étudiés.

La deuxième approche incorpore le contexte temporel dans une étape autre que le post-traitement des sorties de classification. En l'occurrence, celle de l'extraction des descripteurs. Pour détecter un concept c dans un plan s_i, cette approche combine les descripteurs extraits des plans voisins, en appliquant une fusion précoce sur les descripteurs appartenant à une fenêtre temporelle centrée sur s_i et englobant des plans successifs précédant et d'autres suivant le plan s_i. Chacun des segments temporels concaténés est pondéré par un poids qui lui est attribué, de manière à donner plus d'importance aux plans qui sont plus proches du plan courant ; le poids de ce dernier sera le plus important. Malgré une utilisation peu optimale, cette méthode a amélioré un système de référence, mais elle s'est montrée moins efficace que le re-scoring temporel. Cependant, nous prévoyons de meilleures performances des descripteurs temporels avec une optimisation de la taille de la fenêtre temporelle et des poids des différents segments temporels considérés.

Ces deux approches ont montré l'importance et l'utilité du contexte temporel pour l'indexation sémantique des vidéos. Après avoir démontré l'importance du contexte sémantique dans le chapitre 3, il devient crucial et important d'étudier des approches combinant les deux types de contexte sémantique et temporel, pour améliorer la performance d'un système de détection de concepts dans les vidéos. Cela est l'objet du chapitre suivant.

Chapitre 5

Contributions pour l'utilisation conjointe des contextes sémantique et temporel

Nous avons vu dans les chapitres précédents que les contextes sémantique et temporel s'avèrent utiles pour améliorer la performance des systèmes d'indexation des images et vidéos. Il serait intéressant de combiner ces deux types de contexte pour l'indexation des vidéos. Nous présentons dans ce chapitre deux approches qui consistent à combiner deux méthodes utilisant chacune un type différent de contexte : le contexte sémantique ou temporel, pour la détection de concepts visuels dans les vidéos. La première combine les deux méthodes "ré-ordonnancement sémantique basé sur une ontologie" présentée dans la section 3.2 et "le re-scoring temporel" décrite dans la section 4.1. Dans la deuxième contribution, nous proposons d'incorporer l'aspect temporel dans l'approche de "rétroaction conceptuelle" présentée dans la section 3.4.

5.1 Re-scoring à deux couches

5.1.1 Description de l'approche

Nous proposons de combiner le ré-ordonnancement sémantique basé sur une ontologie, et précisément la méthode "Famille de concept" présentée dans la section 3.2 et l'approche du "re-scoring temporel" décrite dans la section 4.1. Plusieurs possibilités sont possibles pour combiner ces deux méthodes. La manière la plus simple est d'utiliser chacun des deux contextes séparément, puis fusionner les deux résultats obtenus. Nous proposons une approche à deux couches : "ré-ordonnancement à deux couches", où chacune des deux méthodes est appliquée dans une couche.

L'approche "ré-ordonnancement à deux couches" consiste à appliquer un premier traitement dans une première couche, puis appliquer un second traitement sur les résultats de la première couche. Nous choisissons de combiner les deux approches : "Famille de concept" et "re-scoring temporel". Parce qu'il est difficile

de prévoir *a priori* quelle est la meilleure façon de combiner les deux approches, nous proposons trois combinaisons possibles, décrites dans la figure 5.1 :

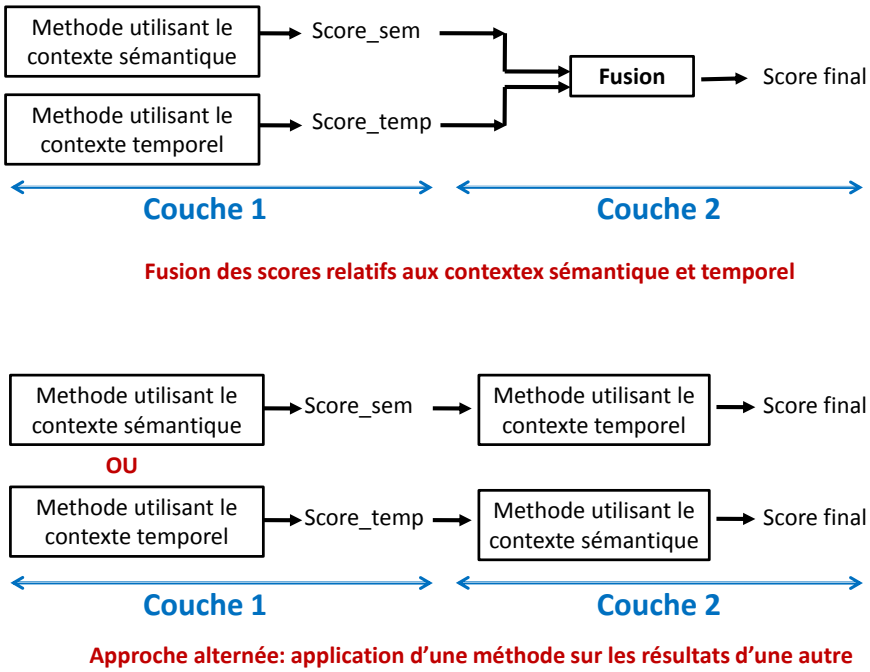


FIGURE 5.1 – Différentes possibilités de combinaison du contexte sémantique et le contexte temporel : Approches à deux couches.

1. Fusion : fusion des résultats des deux approches en moyennant pour chaque plan vidéo, les scores de détection obtenus par les deux méthodes :

$$F_{ac}(e, c_i) = (F_{ac-s}(e, c_i) + F_{ac-t}(e, c_i))/2 \tag{5.1}$$

   où $F_{ac}(e, c_i)$ est le nouveau score de détection de c_i dans le plan e, $F_{ac-s}(e, c_i)$ et $F_{ac-t}(e, c_i)$ représentent les scores de détection de c_i dans le plan e en utilisant "Famille de concept" et "re-scoring temporel", respectivement ;
2. Application de la méthode " re-scoring temporel" sur les résultats de l'approche "Famille de concept" ;
3. Application de l'approche "Famille de concept" sur les résultats de la méthode " re-scoring temporel" ;

5.1.2 Expérimentations et résultats

Nous avons testé et évalué notre approche décrite précédemment dans le contexte de la tâche d'indexation sémantique de TRECVid 2010. Ces expérimentations sont une extension de celles effectuées dans le cadre de l'approche "ré-ordonnancement sémantique basé sur une ontologie" présentée dans la section 3.2. Nous avons utilisé le même protocole expérimental en considérant les mêmes descripteurs et les mêmes types de classificateur et méthode de fusion. Pour rappel, nous avons considéré :

- Cinq descripteurs individuels : *hist*, *gab*, *hg*, *sift*, *audio* ;
- Fusion tardive des cinq descripteurs individuels. Ce résultat sera appelé : *d_fusion* ;
- Fusion tardive des résultats de 40 descripteurs individuels. Nous appelons ce résultat :*Quaero_fusion* ;

Nous avons utilisé les MSVM et les KNNC [1] comme classificateurs. Nous avons fusionné pour chacun des trois cas cités précédemment les résultats de ces deux classificateurs pour améliorer la performance et gagner en robustesse. Vous pouvez vous référer à la section 3.2.2 pour plus de détails sur le protocole expérimental.

Nous avons appliqué ensuite notre approche de ré-ordonnancement à deux couche avec les trois combinaisons proposées, sur les différents résultats précédemment cités : cinq descripteurs individuels, *d_fusion* et *q_fusion*. Pour l'approche du "re-scoring temporel", nous avons utilisé une fenêtre de voisinage temporel de taille égale à 11 plans (Les 5 plans successifs précédant et les 5 plans successifs suivant le plan courant). Ce choix a été déterminé par validation croisée. En effet, les meilleurs résultats sont obtenus pour une taille $(2 * w + 1)$ allant de 7 $(w = 3)$ à 11 plans $(w = 5)$.

Nous utilisons dans ce qui suit les notations suivantes :

- *cF* : résultat de l'approche "Famille de concept" seule, sans combinaison avec le re-scoring temporel ;
- *2lay_Fu* : fusion (moyenne) des scores obtenus par les deux approches : " re-scoring temporel" et "Famille de concept" ;
- $T \rightarrow cF$: application de la méthode " re-scoring temporel" sur les résultats de l'approche "Famille de concept" ;
- $cF \rightarrow T$: application de l'approche "Famille de concept" sur les résultats de la méthode " re-scoring temporel" ;

Résultats :

Le tableau 5.1 présente les résultats en termes de précision moyenne inférée (infAP) du ré-ordonnancement à deux couches sur les différents descripteurs et fusions considérés, et ce, pour la fusion tardive des résultats de MSVM et KNNC. Le ré-ordonnancement à deux couches améliore les résultats quelque soit le type de descripteur utilisé. Cependant *2lay_Fu* reste la moins efficace des trois combinaisons. Les meilleurs résultats sont obtenus en appliquant le "re-scoring temporel" sur les résultats de l'approche "Famille de concept", dans ce cas, le gain relatif sur la MAP varie entre 14.96% et 27.20%. La meilleure valeur de MAP est égale à $0,1589$, qui est atteinte en utilisant *Quaero_fusion* comme système de référence. Le gain relatif lors de l'utilisation des scores de Multi-SVMs seuls sans les fusionner avec ceux des KNNC, est compris entre 12,37% et 33,34%. Lors de l'utilisation des résultats de KNNC seuls sans les fusionner avec ceux des Multi-SVMs, l'amélioration relative sur la MAP varie entre 14,62% et 32,06%.

1. Une version de K-plus-proches voisins où les paramètres sont optimisés pour chaque concept indépendament des autres, et où les votes sont pondérés par les distances séparant l'échantillon de ses k-plus proches voisins.

	initial	*cF*	*2lay_Fu*	T → cF	cF → T
hist	0.0343	0.0356 (+3.79%)	0.0399 (+16.33%)	0.0419 (+22.16%)	0.0421 (**+22.74%**)
gab	0.0307	0.0315 (+2.60%)	0.0342 (+11.40%)	0.0353 (+14.98%)	0.0354 (**+15.31%**)
hg	0.0548	0.0559 (+2.01%)	0.0609 (+11.13%)	0.0631 (+15.14%)	0.0630 (**+14.96%**)
sift	0.0698	0.0725 (+3.87%)	0.0789 (+13.04%)	0.0818 (+17.19%)	0.0831 (**+19.05%**)
audio	0.0136	0.0146 (+7.35%)	0.0159 (+16.91%)	0.0169 (+24.26%)	0.0173 (**+27.20%**)
d_fusion	0.0832	0.0856 (+2.88%)	0.0944 (+13.46%)	0.0976 (+17.31%)	0.0986 (**+18.51%**)
q_fusion	0.1428	0.1478 (+3.50%)	**0.1563** (+9.45%)	**0.1577** (+10.43%)	**0.1589** (**+11.27%**)

TABLE 5.1 – Les résultats (MAP (gain)) de l'approche de ré-ordonnancement à deux couches. L'évaluation à été réalisée sur TRECVid 2010.

Pour résumer, les performances du système lors de l'application du reclassement sur les scores de MSVM sont meilleures que lors de l'utilisation des résultats de KNNC, mais le gain est plus élevé pour KNNC. De plus, les meilleures performances sont obtenues en appliquant le ré-ordonnancement sur les scores de la fusion tardive. Néanmoins, le gain est plus faible que lors de l'utilisation des résultats de chaque classificateur séparément. Nous pouvons expliquer cela par le fait que plus les résultats initiaux sont bons, plus il est difficile de les améliorer.

D'autre part, nous pouvons remarquer que chacune des trois combinaisons des contextes sémantique et temporel donne de meilleurs résultats que l'application de la méthode "Famille de concept" seule. Cela confirme notre hypothèse de départ et confirme l'utilité et l'importance de l'exploitation des deux types de contexte pour l'indexation sémantique des vidéos.

5.2 Rétroaction conceptuo-temporelle

5.2.1 Description de l'approche

Nous proposons d'incorporer le contexte temporel dans l'approche "Rétroaction conceptuelle" présentée dans la section 3.4 du chapitre 3. Un descripteur de haut niveau d_i est construit comme décrit dans la méthode "Rétroaction conceptuelle" classique. Nous appliquons une fusion précoce des descripteurs conceptuels d'un certain nombre de plans voisins, pour générer un nouveau descripteur de haut niveau d'_i pour le plan (i). d'_i est le résultat de concaténation des descripteurs conceptuels des plans voisins consécutifs appartenant à une fenêtre

temporelle centrée sur le plan courant (i) et de taille égale à $2 \times w + 1$: le plan courant i, les w plans précédents et les w plans suivants.

Le descripteur d'_i est utilisé par un nouveau classificateur pour prédire de nouveaux scores de détection. Ces scores sont fusionnés avec ceux renvoyés par le premier apprenant entrainé sur des descripteurs de bas niveaux comme décrit dans l'approche classique de rétroaction. La figure 5.2 décrit ce processus global.

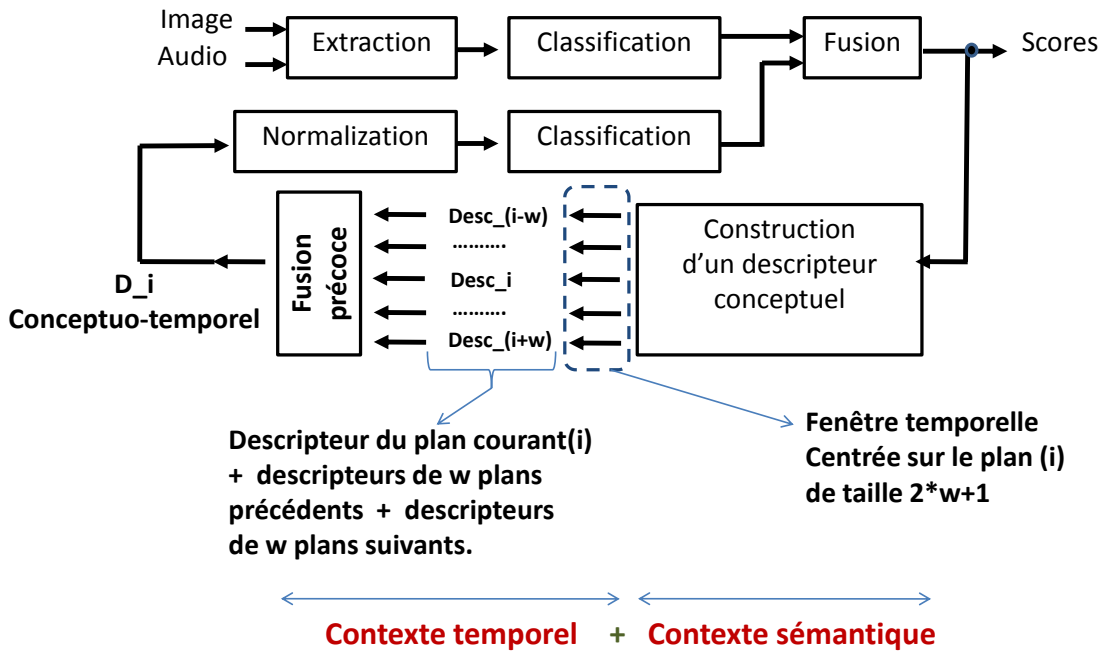


FIGURE 5.2 – Rétroaction conceptuo-temporelle.

Intuitivement, plus deux plans vidéo sont chronologiquement proches, plus leurs contenus sont liés visuellement et sémantiquement. En effet, si un plan p_i contient un concept cible c alors le plan p_{i+1} qui le suit et le plan p_{i-1} qui le précède ont plus de chances de contenir également le concept c. Et plus on s'éloigne du plan p_i la probabilité d'apparition du concept c dans les plans diminue. En se basant sur ce raisonnement, nous proposons de pondérer les descripteurs des plans voisins concaténés des poids inversement proportionnels aux distances les séparant du plan courant. Ainsi, plus un plan est proche du plan courant, plus son poids associé est grand. De ce fait, un descripteur conceptuel avant la fusion précoce n'est pas constitué uniquement de scores, comme dans l'approche de rétroaction conceptuelle classique, mais des scores de détection pondérés par le poids associé au plan en question. La figure 5.3 décrit ce processus.

La rétroaction conceptuo-temporelle peut être utilisée de façon itérative comme présenté dans la section 3.6 pour les approches classique et étendue.

5.2.2 Expérimentations et résultats

Nous avons testé et évalué notre approche dans le contexte de la tâche d'indexation sémantique (voir la section 2.13.2.3) de TRECVid 2012 [OAM+12]. Nous

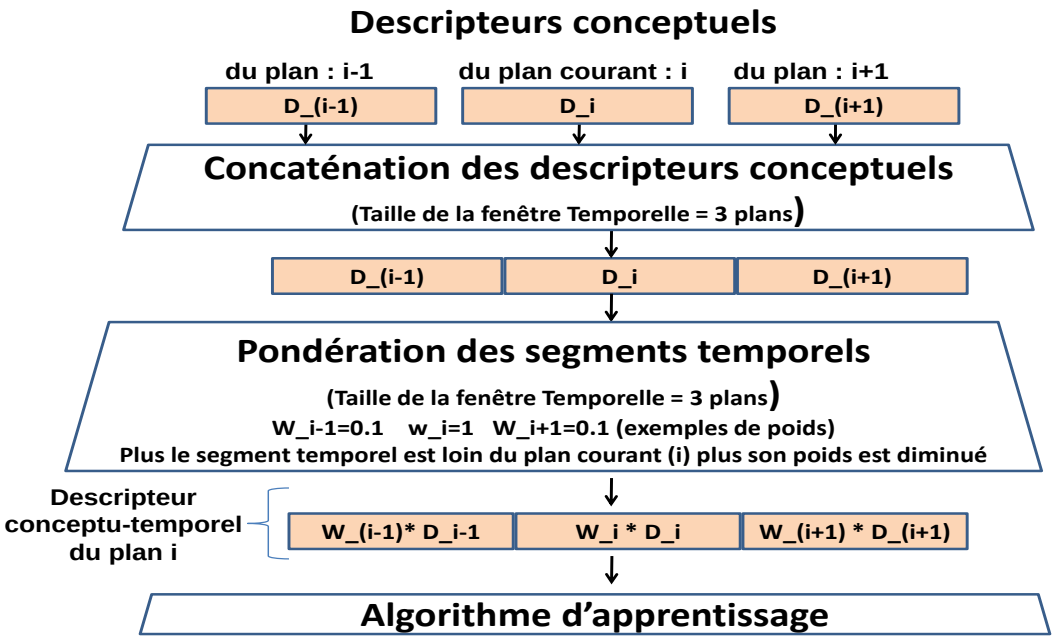


FIGURE 5.3 – Pondération des fragments temporels dans l'approche de rétroaction conceptuo-temporelle.

utilisons le même protocole expérimental décrit pour la rétroaction conceptuelle classique : le même corpus de données, les mêmes annotations, les mêmes descripteurs, les mêmes types de classificateurs et le même schéma de fusion. Vous pouvez vous référer à la section 3.4.2 du chapitre 3 pour plus de détails sur le protocole expérimental.

La mesure d'évaluation est la précision moyenne inférée (infAP) sur les 46 concepts évalués, qui est une estimation classique de la mesure MAP obtenue en utilisant la méthode de Yilmaz et al. [YA06].

Les résultats sont présentés dans ce qui suit sur l'ensemble de test uniquement, mais le réglage et l'optimisation des paramètres sont effectués par validation croisée sur le corpus de développement. Les résultats sur le corpus de test sont consistants avec ceux de l'ensemble de développement.

Nous avons appliqué ensuite notre approche "Rétroaction conceptuo-temporelle", en utilisant les scores renvoyés par le système de référence. Nous rappelons qu'à ce stade, nous appliquons l'approche classique la rétroaction conceptuelle en suivant le même protocole expérimental qui est décrit dans la section 3.4.2 du chapitre 3, où la rétroaction est appliquée après le re-scoring temporel, en prenant touts les concepts, sans filtrage de concepts ni pondération par des coefficients de corrélation.

Nous avons considéré pour le contexte temporel une fenêtre de taille $T = 3$ ($T = 2 \times w + 1$, donc $w = 1$) , pour les mêmes raisons évoquées lors de la présentation de l'approche "Descripteurs temporels" dans la section 4.2. La dimension du nouveau descripteur conceptuel devient $3 \times 346 = 1038$. Pour les

poids, nous avons utilisé $w_i = 1$ pour le plan courant, et une optimisation globale (pour tous les concepts et non pas par concept) des poids des plans voisins a été réalisée sur un corpus de développement. Plus un plan est proche du plan courant plus son poids est plus important (< 1).

Nous avons considéré pour l'adaptation de la fenêtre temporelle trois cas :

1. Uniquement $2 \times w$ plans précédant le plan courant ;
2. Uniquement $2 \times w$ plans suivant le plan courant ;
3. Les w plans précédents et les w suivants ;

Nous avons construit ensuite des détecteurs des 346 concepts, pour ces trois cas. Nous avons fusionné après les résultats obtenus à cette étape avec ceux du système de référence initial (fusion tardive des descripteurs de bas niveau), en moyennant leurs scores respectifs pour chaque plan vidéo. Nous avons appliqué après le re-scoring temporel pour la génération des scores de détection finaux.

Nous utiliserons dans ce qui suit les notations suivantes :

- fb_tmp : rétroaction conceptuo-temporelle, en considérant les w plans précédents et les w suivants ;
- fb_prev : rétroaction conceptuo-temporelle, en considérant uniquement $2 \times w$ plans précédant le plan courant ;
- fb_foll : rétroaction conceptuo-temporelle, en considérant uniquement $2 \times w$ plans suivant le plan courant ;
- fb : rétroaction conceptuelle classique. Cette méthode a été présentée dans la section 3.4 du chapitre 3 ;
- $fb \times corrS$: rétroaction conceptuelle étendue avec pondération des dimensions conceptuelles par des coefficients de corrélations entre les détecteurs de concepts. Cette méthode a été présentée dans la section 3.5.1 du chapitre 3 ;
- Signal : fusion tardive des descripteurs de bas niveau ;
- Concept : sorties d'un classificateur entrainé sur un descripteur conceptuel ;
- Signal + Concept : fusion tardive "Signal" et "Concept" ;
- Signal + Concept+ TRS : résultat de l'application du re-scoring temporel sur les résultats de " Signal + Concept" ;

Les résultats présentés ci-après sont atteints en prenant comme poids : i) fb_tmp : 1 pour $plan_i$, 0.05 pour $plan_{i-1}$ et $plan_{i+1}$; ii) fb_foll : 1 pour $plan_i$, 0.05 pour $plan_{i+1}$ et 0.01 pour $plan_{i+2}$; iii) fb_prev : 1 pour $plan_i$, 0.05 pour $plan_{i-1}$ et 0.01 pour $plan_{i-2}$; où $plan_i$ est le plan courant. Ces valeurs sont déterminées par validation croisée.

Résultats :

Le tableau 5.2 présente les résultats obtenus par l'approche "rétroaction conceptuo-temporelle" en termes de précision moyenne inférée (infAP) et le gain relatif sur infAP. Notre méthode améliore la performance du système initial en atteignant un gain relatif variant entre 5.5% (MAP=0.2840) et +12.7% (MAP=0.3032). Cette approche est meilleure que l'approche de "rétroaction conceptuelle" classique, mais reste moins efficace que "$fb \times corrS$" qui donne de meilleurs résultats sauf à la sortie de l'étape de classification où les résultats sont

légèrement meilleurs en faveur de l'approche "rétroaction conceptuo-temporelle". Cette observation peut être expliquée par le fait que le paramètre w n'a pas été optimisé finement, pour la simple raison qu'une grande valeur de w engendre une explosion de la taille du descripteur résultant. Dans le cas de descripteurs de petites tailles, nous prévoyons de meilleures performances, surtout avec une optimisation locale (pour chaque concept séparément). D'autre part, nous pensons avoir introduit du bruit en considérant l'aspect temporel à plusieurs étapes : en appliquant le re-scoring temporel et pendant la création du descripteur conceptuo-temporel. En effet, contrairement à la rétroaction conceptuo-temporelle, dans $fb \times corrS$, la dimension temporelle est exploitée une seule fois via le re-scoring temporel. Ainsi, beaucoup moins de bruit est introduit dans cette méthode. À cause de cette même raison, l'application du TRS sur le résultat de la rétroaction conceptuo-temporelle détériore les performances. Nous constatons également que les plans voisins qui précédent le plan courant aident à mieux détecter les concepts que ceux qui le suivent, comme le montrent les résultats obtenus par fb_prev. La différence de performances entre l'utilisation des plans précédant et les plans suivant le plan courant n'est pas significative, mais Il serait intéressant d'étudier la variation de cette différence de performance dans le cas où w est optimisé localement. Centrer la fenêtre sur le plan courant (prendre des plans précédant et suivant le plan courant) est la stratégie la plus efficace, comme le montrent les résultats.

Table 5.2 – Résultats (infAP) de l'approche "rétroaction conceptuo-temporelle".

	fb	$fb \times corrS$	fb_tmp	fb_prev	fb_foll
Concept	0.2644	0.2852	0.2874	0.2867	0.2840
Signal + Concept	0.2925	0.3068	0.3032	0.3023	0.3021
Signal + Concept + TRS	0.2981	0.3082	0.3025	0.3016	0.3010
système de référence	0.2691				

La figure 5.4 montre une comparaison entre les résultats en termes de précision moyenne (AP) obtenus pour certains concepts par les approches suivantes : 1) Rétroaction conceptuelle classique (voir la section 3.4), 2) Rétroaction conceptuelle étendue (pondération des dimensions conceptuelles par les corrélations inter-détecteurs de concept, voir la section 3.5.1) et 3) Rétroaction conceptuo-temporelle. Nous avons choisi de détailler les résultats de " Signal + Concept + TRS". Les valeurs de AP confirment les remarques faites pendant l'analyse de MAP. En effet, nos différentes propositions améliorent le système de référence qui est déjà bon en termes de performance (précision), pour l'ensemble des concepts. L'extension "rétroaction conceptuo-temporelle" améliore le système initial dans tous les cas et pour tous les concepts, mais reste moins efficace que "$fb \times corrS$". Nous pensons que cela est dû à l'accumulation du bruit introduit par l'aspect temporel. En effet, l'information temporelle est incorporée dans le descripteur conceptuo-temporel via TRS avant la rétroaction, pendant la construction du descripteur et dans le TRS final.

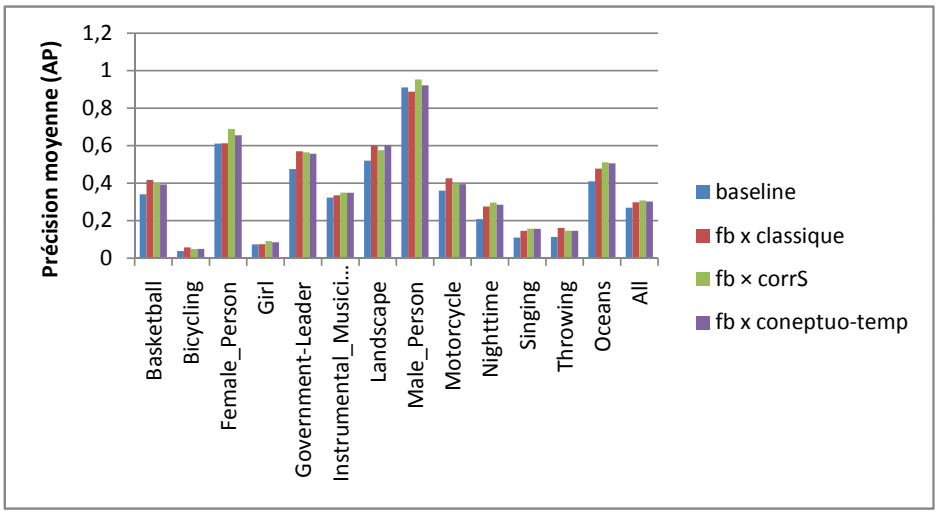


FIGURE 5.4 – Comparaison entre les résultats (précision moyenne) des approches proposées pour certains concepts.

Rétroaction conceptuelle itérative :

Nous avons déroulé les différentes versions de la rétroaction de façon itérative comme présenté dans le sections 3.6. Le tableau 5.3 montre un récapitulatif des résultats d'une exécution itérative dans un cycle de trois itérations consécutives, et compare les résultats des différentes versions de rétroaction.

	approche classique		$fb \times corrS$		fb_tmp	
	par plan	+TRS	par plan	+TRS	par plan	+TRS
iter0	0.2613 -	0.2691 +3.0%	- -	- -	- -	- -
iter 1	0.2925 +11.9%	0.2981 +14.1%	0.3068 +17.4%	0.3082 +18.0%	0.3032 +16%	0.3025 +15.8%
iter 2	0.2984 +14.2%	0.3014 +15.3%	0.2855 +9.3%	0.2862 +9.5%	0.2915 +11.6%	0.2910 +11.4%
iter 3	0.2980 +14%	0.3011 +15.2%	0.2632 +0.7%	0.2674 +2.3%	0.2898 +10.9%	0.2901 +11%

TABLE 5.3 – Résultats de l'approche rétroaction conceptuelle itérative avec re-scoring temporel (TRS) : MAP et % de gain relatif. iter0 correspond au système initial de référence sans appliquer la rétroaction conceptuelle.

On peut voir que l'approche itérative n'est bénéfique que pour l'approche classique. En effet, "$fb \times corrS$" et la rétroaction conceptuo-temporelle, avec TRS, atteignent une performance maximale dans la première itération, avec un gain relatif en MAP d'environ +18.0% et +15.8%, respectivement. Ceci peut être ex-

pliqué par l'accumulation du bruit introduit par la considération d'informations contextuelles issues de sources différentes. Cependant, ces deux méthodes atteignent en une seule itération des résultats comparables (voire meilleurs) à ceux obtenus par la version classique dans la deuxième itération. On conclut donc, que la version classique capture les informations contextuelles progressivement au fur et à mesure des itérations, alors que les deux autres extensions arrivent à capturer l'ensemble des informations contextuelles en une seule fois, via l'exploitation de sources de contextes additionnels.

La performance globale atteinte par l'approche rétroaction conceptuo-temporelle est comparable (légèrement meilleure) à la performance du meilleur système officiellement évalué dans TRECVid 2012 dans les mêmes conditions (0.2978). La Rétroaction conceptuo-temporelle a atteint un gain relatif d'environ +16.0% (MAP=03032) sans TRS.

5.3 Conclusion

Nous avons présenté dans ce chapitre deux approches combinant les deux contextes sémantique et temporel pour l'indexation des vidéos. La première proposition combine deux méthodes qui exploitent chacune un des deux types de contexte séparément dans deux couches : "re-ordonnancement sémantique" présentée dans la section 3.2 du chapitre 3 et "re-scoring temporel" décrite dans la section 4.1 du chapitre 4. Dans la première couche, les scores d'une première classification sont modifiés en utilisant des informations contextuelles. Dans la seconde couche, les résultats du premier traitement sont exploités pour calculer de nouveaux scores de détection des concepts. Les résultats ont montré que pour cette stratégie, il vaut mieux appliquer le re-soring temporel sur les résultats d'un re-scoring sémantique. Bien que cette méthode améliore les résultats, elle reste vulnérable à la propagation des erreurs et sujette au problème du sur-apprentissage, et surtout au problème de normalisation des scores.

La deuxième contribution consiste à exploiter de l'information temporelle et de l'incorporer dans l'approche de rétroaction conceptuelle qui utilise le contexte sémantique et qui a été présentée dans le chapitre 3. Cette proposition consiste à construire des descripteurs conceptuels à partir des scores de détection de concepts et d'appliquer pour chaque plan une fusion précoce de ces descripteurs appartenant à une fenêtre temporelle centrée sur le plan courant. Ces nouveaux descripteurs conceptuo-temporels sont ensuite utilisés par un apprenant de la même manière qu'un descripteur de bas niveau. Contrairement à la proche du re-scoring à deux couches, la rétroaction conceptuo-temporelle considère une phase d'apprentissage. Cela constitue un avantage majeur par rapport à l'approche à deux couche, car en utilisant un apprenant le problème de normalisation des scores est contourné. D'une part, le gain atteint par la rétroaction conceptuelle est plus important que celui obtenu par le re-scoring à deux couches qui souffre beaucoup plus du handicap de propagation des erreurs d'une étape à celle qui la suit.

Chapitre 6

Contribution pour la détection simultanée de groupes de concepts dans les documents multimédia

Nous présentons dans ce chapitre nos contributions pour la détection simultanée de groupes de concepts (multi-concept) visuels dans les documents multimédia. Nous rappelons que cette tâche est liée à la problématique de l'utilisation du contexte pour l'indexation des documents multimédia. Nous considérons que détecter simultanément un groupe de concepts revient à détecter un concept ou plusieurs concepts dans un contexte ou les autres sont présents. Nous avons vu dans la section 2.12 qu'il existe deux catégories d'approches pour détecter simultanément un groupe de concepts (multi-concept) dans les documents multimédia. À titre de rappel, pour détecter un groupe de concepts dans un document multimédia, une méthode consiste à construire un modèle spécifique pour chaque multi-concept. Nous adoptons dans notre travail ce type d'approche en utilisant une méthode d'apprentissage d'ensembles (Ensemble learning), avec une philosophie similaire à celle de [LSWS12]. Une alternative à cette idée consiste à détecter les concepts individuels formant le groupe séparément et fusionner ensuite les résultats de leurs détecteurs respectives. Nous allons présenter dans ce qui suit une étude comparative entre ces deux méthodes.

6.1 Description des approches

6.1.1 Modèles de concepts multiples

Nous utilisons une méthode qui consiste à générer un modèle spécifique pour chaque groupe de concepts. Contrairement à [LSWS12], nous ne considérons pas de données supplémentaires, mais nous nous servons uniquement des données relatives aux concepts singuliers.

En se basant sur les notations présentées dans la section 3.1, nous modélisons cette approche par un 7-tuple :

$< C, C^{multi}, E(E_D \cup E_T), A, A^{multi}, F_{descr}, F_{sc}^{multi} >$

Où :

- C^{multi} : est l'ensemble de concepts multiples. $C^{multi} \subset \mathcal{P}(C)$, où $\mathcal{P}(C)$ représente l'ensemble de tous les sous-ensembles de C. Chaque multi-concept est un ensemble de concepts singuliers. $|c_{multi}| = 2$ représente le cas des paires de concepts, $|c_{multi}| = 3$ concerne celui des triplets de concepts, ainsi de suite ;
- $A^{multi} : E_D \times C^{multi} \rightarrow \{-1, 1, 0\}$; est une fonction qui renvoie une valeur entière correspondant à l'annotation d'un échantillon $e \in E_D$ par un multi-concept $c_{multi} \in C^{multi}$; avec : $A^{multi}(e, c_{multi}) = 1$ signifie que l'échantillon est annoté positif et $A^{multi}(e, c_{multi}) = -1$ signifie qu'il est étiqueté négatif, et $A^{multi}(e, c_{multi}) = 0$ concerne le cas où e n'est pas annoté ;
- $F_{sc}^{multi} : E \times C^{multi} \rightarrow R$, est une fonction de décision qui renvoie une valeur correspondant à la sortie d'un détecteur d'un multi-concept ;

Pour instancier ce modèle, nous définissons dans ce qui suit les fonctions A^{multi} et F_{sc}^{multi}. Nous nous basons pour cela uniquement sur les données relatives aux concepts singuliers.

Étant donné un ensemble d'échantillons annoté par un ensemble de concepts singuliers c_i, nous générons les annotations des mêmes échantillons par un multi-concept c_{multi} en réalisant une intersection des annotations par les concepts composant c_{multi} :

$$A^{multi}(e, c_{multi}) = \begin{cases} 1 & Si \; \forall \; c \; \in c_{multi} : A(e, c) = 1 \\ -1 & Si \; \exists \; c \in c_{multi} : A(e, c) = -1 \\ 0 & Sinon \end{cases} \tag{6.1}$$

Il y a généralement peu d'exemples positifs et beaucoup d'exemples non annotés. Par conséquent, la fonction A^{multi} donne une matrice d'annotations creuse. Les échantillons non annotés ne seront pas utilisés dans l'étape d'apprentissage. Ce phénomène complique la situation et affecte négativement la performance des classificateurs. Pour pallier cet inconvénient, nous proposons d'utiliser comme fonction F_{sc}^{multi}, une méthode d'ensemble pour l'apprentissage et plus précisément le "Bagging". Ce genre d'approches est compatible avec le problème de classes déséquilibrées. Nous choisissons comme classificateur, "MSVM" qui est décrit dans [SQ10] (voir la section 2.7.4.2), pour ses bons résultats dans le contexte de la détection de concepts visuels dans les vidéos et sa compatibilité avec le problème de classes déséquilibrées.

Pour décrire un plan vidéo (F_{descr}), nous utilisons plusieurs types de descripteurs pour chaque plan vidéo. F_{descr} correspond donc, à la méthode d'extraction de descripteurs de bas niveau (e.g. Visuel, audio, etc) des plans vidéo. Nous aborderons les détails concernant les descripteurs utilisés dans la section 6.2. Pour la fusion tardive des résultats obtenus par les différents descripteurs utilisés, nous choisissons d'utiliser une somme pondérée où les poids sont des scores

de confiance :

$$F_{sc}^{multi}(e, c_{multi}) = \frac{1}{nbC} \times \sum_{i=1}^{nbC} F_{sc}^{multi}(e_i, c_{multi}) \times Conf_i \tag{6.2}$$

où e_i est l'échantillon e représenté par le i^{eme} descripteur, $F_{sc}^{multi}(e_i, c_{multi})$ est le score obtenu par le i^{eme} résultat (en utilisant le i^{eme} descripteur), $Conf_i$ est un score de confiance concernant le i^{eme} résultat, et nbC est le nombre de descripteurs utilisé. Nous choisissons la précision moyenne calculée sur un corpus de développement comme score de confiance. Le score renvoyé par $F_{sc}^{multi}(e, c_{multi})$ sera le score final de détection simultanée du groupe de concepts c_{multi} dans l'échantillon e.

Nous appellerons cette méthode dans ce qui suit : $learnMulti$.

6.1.2 Fusion de détecteurs de concepts individuels

Les méthodes qui consistent à fusionner les scores de détection ont un grand avantage sur la méthode d'apprentissage directe : elles ne nécessitent pas de lourds traitements. Les méthodes que nous allons utiliser n'ont aucun paramètre à apprendre et/ou à optimiser, ce qui permet leur application directement et facilement sur n'importe quel groupe de concepts. En se basant sur les notations présentées dans la section 3.1, nous proposons le modèle général suivant :

$< C, C^{multi}, E(E_D \cup E_T), F_{descr}, A, F_{sc}, F_{sc}^{multi} >$

Où :

- $F_{sc}^{multi} : E \times C^{multi} \rightarrow R$. Cette fonction détecte un multi-concept $c_{multi} \in C^{multi}$ dans un échantillon $e \in E$ et renvoie une valeur correspondant au score de détection ;
- Les autres paramètres ont déjà été définis dans le modèle de la section 6.1.1.

Pour instantier le modèle nous définissons F_{sc} et F_{sc}^{multi}. Nous choisissons comme fonction F_{sc} la même méthode choisie pour la génération de modèle pour les concepts multiples : "MSVM", pour les mêmes raisons de performance. La seule différence est qu'à ce stade on utilise comme annotations, le résultat de la fonction A au lieu de A^{multi}, qui dans notre cas est remplacée par un expert humain fournissant des annotations manuelles.

En ce qui concerne la fonction F_{sc}^{multi}, nous allons examiner plusieurs méthodes de fusion de scores. Nous avons fait dans nos expériences le maximum pour rendre ou garder ces scores de probabilités homogènes. Cela peut être fait en appliquant la normalisation de Platt (voir la secton 2.7.6), sur les sorties des classificateurs, et/ou en utilisant une fonction de fusion appropriée. En effet, cette normalisation a un effet sur l'efficacité des méthodes envisagées. Nous proposons d'utiliser les méthodes de fusion suivantes :

- Une fusion linéaire, que nous appellerons *linFus* : moyenne arithmétique des scores. Ce type de fusion est listé dans [LSWS12] :

$$F_{sc}^{multi}(e, c_{multi}) = \frac{\sum_{c \in c_{multi}} F_{sc}(e, c)}{|c_{multi}|} \tag{6.3}$$

– Une méthode basée sur la notion de probabilité, appelée dans ce qui suit *prodFus*, ou moyenne géométrique des scores. Cette méthode considère que les scores sont des probabilités et que ces probabilités sont obtenues par des détecteurs conditionnellement indépendants (la racine carrée ne change pas l'ordre des plans/images) :

$$F_{sc}^{multi}(e, c_{multi}) = \sqrt[|c_{multi}|]{\prod_{c \in c_{multi}} F_{sc}(e, c)} \tag{6.4}$$

– Une approche inspirée de l'approche booléenne étendue de [SFW83], que nous nommons *boolFus*, qui considère un multi-concept comme la conjonction des concepts le composant :

$$F_{sc}^{multi}(e, c_{multi}) = 1 - \sqrt{\frac{\sum_{c \in c_{multi}} (1 - F_{sc}(e, c))^2}{|c_{multi}|}} \tag{6.5}$$

Le score renvoyé par $F_{sc}^{multi}(e, c_{multi})$ sera le score final de détection du multi-concept c_{multi} dans le plan e.

6.2 Expérimentations et résultats

6.2.1 TRECVid

Données :

Nous avons testé et évalué les approches décrites pour le cas des paires de concepts (bi-concept) dans les sections précédentes dans le cadre de la sous-tâche "détection de paire de concepts" de la tâche d'indexation sémantique de TRECVid 2013. Les annotations pour les concepts singuliers ont été fournies par l'annotation collaborative de TRECVid [AQ08a]. Cette sous-tâche définit 10 paires de concepts rares et la performance est mesurée par la précision moyenne inférée (InfAP).

Nous avons généré les annotations par paire de concepts à partir de celles des concepts singuliers, comme décrit dans la section 6.1.1 (via la fonction A^{paire}). Le nombre d'exemples positifs résultant est très petit, spécialement pour certains bi-concepts. Le tableau 6.1 montre les paires de concepts utilisées ainsi que des détails sur leurs fréquences dans le corpus d'apprentissage utilisé. Nous pouvons voir que quatre parmi les dix paires de concepts ont moins de 10 exemples positifs. Nous notons aussi que pour faire la validation croisée, ce corpus d'apprentissage doit être divisé en sous-parties, ce qui diminuera encore plus le nombre d'échantillons à l'entrée des classificateurs. Cela reflète la difficulté de la tâche.

Pour la description du contenu des plans vidéo, nous avons utilisé au total de 66 variantes de 12 types de descripteurs fournis par le groupe IRIM [BLS+12]. Les variantes d'un même descripteur se différencient par de légers détails comme par exemple, la taille du dictionnaire utilisé pour la génération des sacs de mots.

Paires de concepts	# positifs	# negatifs	#non annotés
Telephones+Girl	1	18918	527004
Kitchen+Boy	12	43845	502066
Flags+Boat_Ship	4	10123	535796
Boat_Ship+Bridges	27	39200	506696
Quadruped+Hand	26	18392	527505
Motorcycle+Bus	9	82490	463424
Chair+George_Bush	41	23881	522001
Flowers+Animal	67	41875	503981
Explosion_Fire+Dancing	0	19550	526373
Government_Leader+Flags	321	12828	532774

TABLE 6.1 – Fréquences des paires de concepts dans le corpus d'apprentissage TRECVid 2013. Les annotations sont le résultat d'une intersection des annotations des deux concepts singuliers.

Nous avons considéré des descripteurs de type : transformation Gabor pour la texture, histogrammes de couleurs, SIFT, STIP, VLAT, Percepts, etc. Tous ces descripteurs sont répertoriés et décrits dans [BLS+12] et aussi dans l'annexe B.

Pour la détection des concepts singuliers, nous avons utilisé un classificateur supervisé de type MSVM [SQ10]. Comme entrée au MSVM, les descripteurs cités ci-dessus sont extraits de chaque plan vidéo. Pour chaque plan vidéo, les scores obtenus par les différents descripteurs sont fusionnés en utilisant une méthode de fusion hiérarchique décrite dans [TSBB+12], pour donner un score final de détection d'un concept singulier dans le plan vidéo concerné. Finalement les différentes méthodes décrites dans la section 6.1.2 sont appliquées en se basant sur les scores finaux de détection de concepts singuliers précédemment calculés.

Pour l'approche consistant à générer un modèle par paire de concepts ("learnMulti"), les détecteurs sont construits en suivant globalement le même schéma de classification que pour les détecteurs de concepts singuliers (MSVM comme classificateur, même descripteurs). La différence se résume sur les annotations utilisées. Finalement, les résultats obtenus par les différents descripteurs sont fusionnés en utilisant la formule 6.2 présentée dans la section 6.1.1, qui est une moyenne pondérée des sorties des MSVM où les poids sont les valeurs de la précision moyenne calculées sur un corpus de développement. Nous notons qu'à ce stade, nous avons privilégié la somme pondérée sur la fusion hiérarchique [TSBB+12], parce que nous avons constaté que cette dernière ne marche pas bien pour le cas des modèles de paires de concepts, même si ses résultats dans le cas des concepts singuliers sont assez bons. L'ensemble des paramètres d'apprentissage sont optimisés par validation croisée.

Résultats :

Le tableau 6.2 présente les résultats obtenus en termes de précision moyenne inférée InfAP pour les approches considérées. Dans ce tableau, la première ligne présente le meilleur système officiellement évalué dans la sous-tâche "détection

Système	MAP
Meilleure soumission TRECVid 2013	0.1616
linFus	0.1613
prodFus	0.1761
boolFus	0.1724
learnMulti	0.1514

TABLE 6.2 – Résultats sur le corpus de test : InfAP pour les 10 paires de concepts évaluées dans TRECVid 2013.

de paires de concepts" de TRECVid 2013. Le groupe des quatre lignes suivantes concernent les résultats obtenus par les méthodes de fusion et la dernière ligne concerne le résultat obtenu par la méthode d'apprentissage "learnMulti". Les approches de fusion sont nettement plus performantes que celle d'apprentissage. "prodFus" reste la meilleure méthode de fusion. Nous notons dans ce contexte, que dans [LSWS12], les auteurs sont arrivés à une autre conclusion concernant la détection de paires de concepts dans les images fixes. En effet, les auteurs ont conclu que la génération d'un modèle spécifique par paire de concepts est plus efficace que la combinaison des détecteurs de concepts individuels. Or, leurs conditions d'expérimentation diffèrent des nôtres concernant plusieurs points. Premièrement, le type de données considéré dans les deux travaux est différent (vidéos dans notre travail vs. les images fixes dans [LSWS12]). D'autre part, en plus du fait que dans [LSWS12], une collecte de bons exemples positifs et négatifs précède la phase d'apprentissage, nous considérons dans notre travail des paires *rares* de concepts. Nous rappelons comme montré dans le tableau 6.1, que quatre paires de concepts ont moins de 10 échantillons positifs pour l'apprentissage (1 pour Telephones+Girl, 4 pour Flags+Boat_Ship, 9 pour Motorcycle+Bus et 0 pour Explosion_Fire+Dancing). Cela peut expliquer pourquoi la performance est inférieure pour l'approche "learnMulti". Nous notons aussi, que pour certaines paires de concepts pour lesquelles il y a trop peu d'exemples positifs, il s'est avéré impossible de dérouler la méthode d'apprentissage, ce qui a conduit à des résultats médiocres, comme c'est le cas pour les deux paires "Telephones+Girl " et "Explosion_Fire+Dancing" (voir la figure 6.1). Contrairement à notre situation, la fréquence moyenne des paires de concepts utilisées dans [LSWS12] est d'environ 3900, et les auteurs ont focalisé leur effort sur le choix de bons échantillons du web social et sur l'apprentissage. D'autre part, la méthode utilisée pour générer les détecteurs de concepts individuels a certainement aussi un impact sur le résultat de la fusion. En effet, chaque méthode capture des informations sémantiques différentes. Dans [LSWS12], les auteurs disent que la recherche de bi-concepts exige des détecteurs individuels avec une bonne capacité de discrimination, ce qui est le cas dans notre travail. En effet, notre système de détection des concepts singuliers a été classé deuxième dans la tâche d'indexation sémantique (SIN) de la compagne d'évaluation TRECVid 2013.

Nous remarquons aussi dans le même tableau 6.2 que les résultats obtenus dépassent le meilleur résultat officiel de TRECVid 2013, par 9,34 % ($100 \times (0,1767 - 0,1616)/0,1616 = 9,34\%$) de gain relatif. Nous pouvons voir à l'aide du

test *Student* apparié bilatéral que, même si les valeurs de MAP diffèrent beaucoup, les résultats du tableau 6.2 ne sont pas statistiquement différents, avec une valeur de $p < 5$ % , cela est dû principalement au nombre petit de paires de concepts considérées (uniquement 10 paires). Les méthodes présentées ici n'ont pas été toutes incluses dans nos soumissions officielles à la sous-tâche de détection de paires de concepts TRECVid 2013, cela est dû au fait que les résultats définitifs n'étaient pas encore achevés avant la date limite. Malgré cela, nous avons quand même obtenu la quatrième place dans cette compétition.

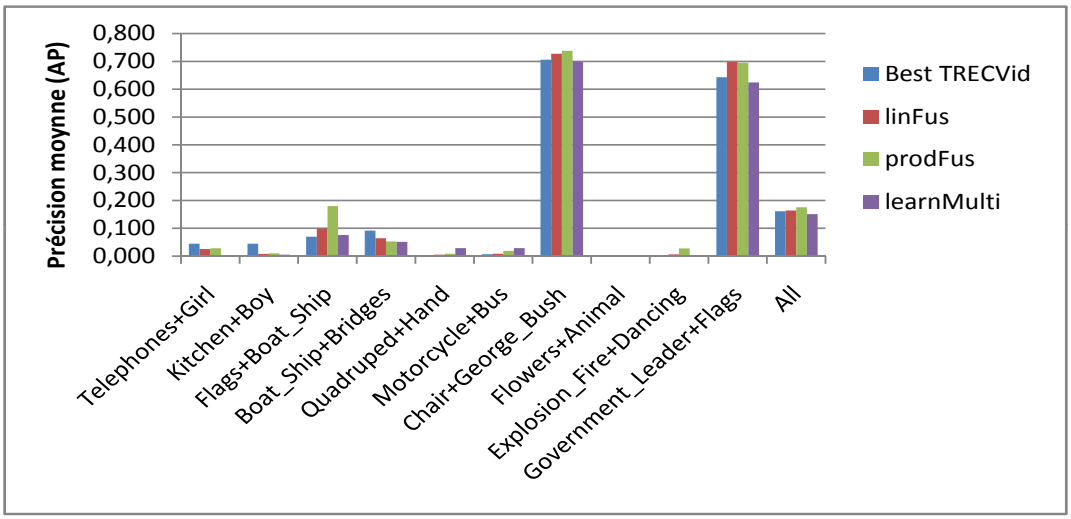


FIGURE 6.1 – Résultats des différentes approches pour la détection de paires de concepts sur la collection TRECVid 2013, et comparaison avec le meilleur résultat de la tâche SIN de TRECVid 2013.

La figure 6.1 décrit les résultats par paire de concepts en termes de précision moyenne (AP) obtenus par les différentes approches considérées et une comparaison avec le meilleur système officiellement évalué dans TRECVid 2013. Nous pouvons voir que l'approche "learnMulti" est la plus efficace pour les paires : "Quadruped+Hand" et "Motorcycle+Bus", mais cette performance ne compense pas l'échec sur les autres paires. Nous constatons aussi, que dans la valeur de MAP globale, il y a les valeurs de AP de deux paires ("Chair+George_Bush" et "Government_Leader+Flags") qui dominent, et que les différentes méthodes sont assez mauvaises en ce qui concerne la détection de certaines paires de concepts. Cela nous motive à utiliser à l'avenir des ressources externes d'informations pour pallier cet inconvénient.

6.2.2 VOC

Nous avons testé et évalué nos approches proposées pour la détection de paires et triplets de concept (bi-concept et tri-concept) dans les images. Nous avons mené

notre évaluation sur la collection Pascal VOC 2012 [1]. Nous avons considéré uniquement les cas de paires et triplet de concepts, à cause de la non-disponibilité de données annotées par des multi-concepts pour des groupes composés d'un plus grand nombre de concepts. Nous avons formé une liste principale de 20 concepts singuliers, en retirant ceux pour lesquels nous n'avions pas assez d'exemples positifs. Nous avons finalement retenu 60 paires et 45 triplets. Malheureusement, nous n'avions pas et ne nous pouvions pas avoir la vérité terrain sur le corpus test concernant les paires et les triplets utilisés. Nous avons donc réalisé, une validation croisée (two-fold cross-validation) en utilisant le corpus de données divisé en deux parties : une pour le développement et l'autre pour la validation. Les annotations des images par les concepts singuliers ont été fournies dans la même collection. Nous avons évalué nos différentes approches proposées en termes de précision moyenne inférée (MAP).

Nous avons généré les annotations par paire et triplet de concepts en utilisant la fonction A^{multi} comme décrit dans la section 6.1.1. Cela a donné, comme attendu, un petit nombre d'exemples positifs. Le nombre d'exemples positifs pour les paires de concepts varie entre 10 et 380, alors que ce nombre varie entre 7 et 98 pour les triplets de concepts. Ce nombre reste quand même raisonnable par rapport au cas de la collection TRECVid comme présenté dans la section 6.2.1. Pour la description du contenu des images, nous avons utilisé sept variantes de trois types de descripteurs : histogramme de couleur, transformation de Gabor et différents variantes de "bags of opponent SIFT".

Nous avons testé ensuite nos approches décrites précédemment pour la détection des paires et triplets de concepts retenus. Pour la méthode basée sur la fusion, nous avons généré en premier, les détecteurs de concepts singuliers en utilisant MSVM comme classificateur supervisé. Comme entrée à ces apprenants, nous avons extrait les descripteurs cités précédemment de chaque image. Nous avons optimisé les descripteurs en utilisant les méthodes : *ACP* et *Power-law* (voir les sections 2.6.2.1 et 2.6.1.1, respectivement). Les sorties des MSVM (en utilisant différents descripteurs) sont finalement fusionnées pour chaque image, via une simple fonction calculant la moyenne des scores de détection. Les trois approches décrites dans la section 6.1.2 sont ensuite appliquées en utilisant ces derniers scores.

Pour l'approche basée sur l'apprentissage d'un modèle par multi-concept, un classificateur de type MSVM est généré pour chaque paire et chaque triplet en utilisant chacun des sept descripteurs considérés et les annotations par paire/triplet. Pour chaque échantillon, les sept résultats sont finalement fusionnés en moyennant les scores pour calculer un score final de détection de chaque paire/triplet dans chaque image.

Résultats :

Le tableau 6.3 présente les résultats en termes de précision moyenne (MAP) obtenus par les différentes approches proposées pour le cas des bi-concepts et

1. Pascal Visual Object Classes (VOC) 2012 collection. http ://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2012/

tri-concepts. Le groupe des trois lignes en haut du tableau décrit les résultats obtenus par l'approche basée sur la fusion de détecteurs de concepts singuliers, et la dernière ligne concerne les résultats de l'approche basée sur l'apprentissage d'un modèle par paire/triplet. Nous pouvons remarquer que pour la détection des bi-concepts, les performances des quatre méthodes sont comparables, avec un léger avantage pour *prodFus* et *boolFus* qui surpassent *learnMulti*, spécialement *prodFus* qui obtient le meilleur résultat. Pour les tri-concepts, *learnMulti* donne des résultats légèrement meilleurs que ceux des méthodes de fusion. Cela confirme la conclusion faite dans [LSWS12] et contrarie le résultat obtenu pour le cas des vidéos, comme présenté dans la section 6.2.1 pour la collection TRECVid. Cependant, cette supériorité n'est pas significative.

	bi-concept	tri-concept
linFus	0.0859	0.0488
prodFus	0.0986	0.0458
boolFus	0.0969	0.0449
learnMulti	0.0920	0.0495

TABLE 6.3 – Résultats (MAP) pour la détection de bi-concepts et tri-concepts sur la collection Pascal VOC'12.

Dans [LSWS12], les auteurs ont évalués leurs contributions en termes de précision dans le top 100, et ont obtenu des valeurs variant entre 14% et 97%. Cependant, ils n'ont considéré que 15 bi-concepts formés par des détecteurs de concepts singuliers qui ont une bonne précision, atteignant pour certains concepts, une précision dans le top 100 égale à 100%. D'autre part, le nombre moyen des paires de concepts utilisées dans [LSWS12] est d'environ 3900, ce qui est différent dans notre cas où un petit nombre d'échantillons positifs est considéré (entre 7 et 380). Cela explique pourquoi les approches décrites dans [LSWS12] paraissaient plus performantes, comparées à nos approches.

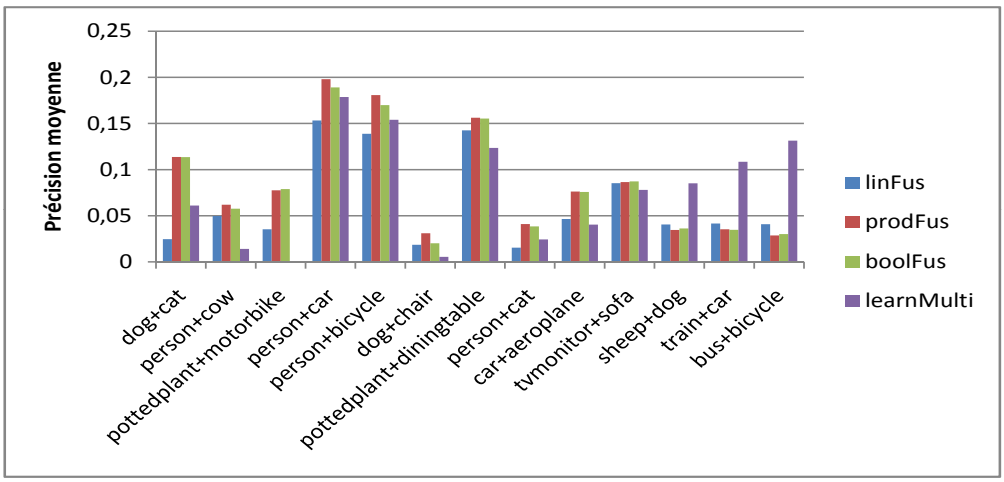


FIGURE 6.2 – Résultats par bi-concept (AP)des différentes approches proposées.

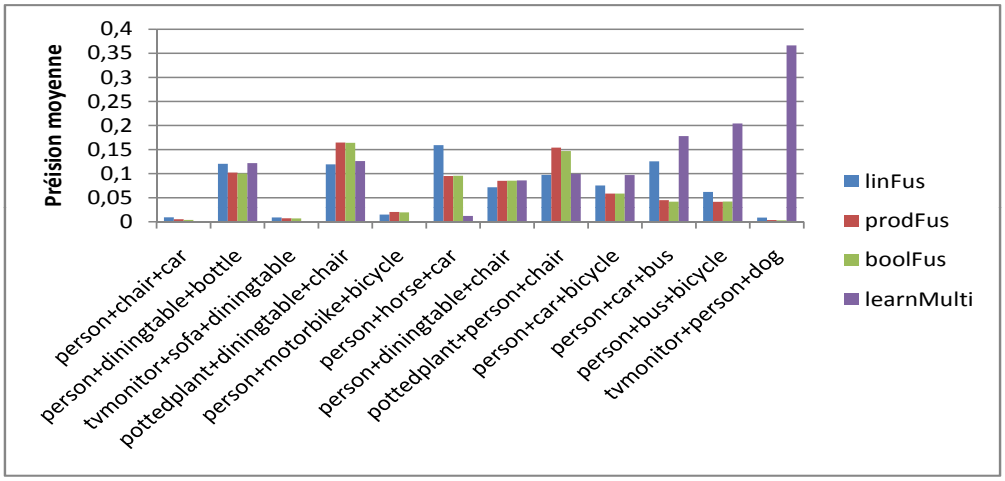


FIGURE 6.3 – Résultats par tri-concept (AP) des différentes approches proposées.

La figure 6.2 décrit les résultats pour certaines paires de concepts en termes de précision moyenne obtenus par les méthodes considérées. Les trois bi-concepts qui sont à droite sont ceux pour lesquels, *learnMulti* donne des résultats nettement meilleurs que ceux atteints par les méthodes basées sur la fusion. Étonnamment, ces trois bi-concepts sont parmi les paires les moins fréquentes dans le corpus de données utilisé, avec un nombre d'échantillons positifs inférieure à 10. D'autre part, pour les deux paires *person+car* et *person+bicycle*, qui font partie des paires les plus fréquentes, avec des centaines d'échantillons positifs, *learnMulti* est moins efficace que les méthode de fusion. Cela peut être expliqué par le principe des classificateurs utilisés. En effet, MSVM utilise un nombre de classificateurs de type SVM qui est inversement proportionnel au nombre d'exemples positifs. Cela signifie que si le nombre d'échantillons positifs est petit, alors il y aura beaucoup plus d'apprenants, et la décision sera donc plus robuste. Une autre remarque importante est que les méthodes basées sur la fusion, spécialement *prodFus*, sont plus efficaces quand les deux concepts formant la paire sont différents ou peu similaires, comme c'est le cas pour *person+cow* et *pottedplant+motorbike*. Par opposition, *learnMulti* est plus performante que les autres approches quand les deux concepts sont liés sémantiquement et/ou sont similaires visuellement, surtout quand ils appartiennent à la même catégorie de concepts, comme par exemple : *bus+bicycle* où les deux concepts font partie de la catégorie de véhicules. Pour la plupart des autres paires, les résultats des deux types d'approches sont comparables.

La figure 6.3 présente les résultats pour certains tri-concepts, en termes de précision moyenne, obtenus par les différentes méthodes considérées. Nous remarquons que la performance est mauvaise pour certains triplets. D'autre part, *learnMulti* obtient des résultats nettement meilleurs que ceux des méthodes basées sur la fusion, surtout ceux qui apparaissent à droite. Là aussi, nous faisons les mêmes remarques que dans le cas des bi-concepts : *learnMulti* est plus efficace pour certains triplets peu fréquents. Nous soulignons que les valeurs de la précision moyenne pour certains triplets, typiquement *tvmonitor+person+dog*, ont dominé les petites valeurs, ce qui explique la supériorité de *learnMulti* par

rapport aux résultats présentés dans le tableau 6.3, dans le cas des tri-concepts. De plus, les méthodes basées sur la fusion de détecteurs de concepts singuliers sont globalement plus efficaces pour les tri-concepts pour lesquels, deux ou trois concepts sont sémantiquement indépendants et/ou peu similaires visuellement (e.g *person+horse+car*). Par opposition, si deux ou trois concepts sont liés sémantiquement ou présentent plusieurs similarités visuelles, *learnMulti* obtient de meilleurs résultats (e.g *bus+car+person*).

6.3 Conclusion

Nous avons comparé deux types d'approches pour la détection simultanée de groupes de concepts dans les images et les vidéos. La première consiste en la construction d'un modèle par groupe de concepts en générant des annotations en se basant sur celles des concepts singuliers, et en utilisant une méthode d'apprentissage d'ensembles. La deuxième est basée sur la combinaison des sorties de détecteurs des concepts singuliers formant le groupe, et ne nécessite aucune étape supplémentaire d'apprentissage. Nous avons instancié et évalué les deux méthodes pour la détection de paires de concepts dans les vidéos, et les paires et les triplets de concepts dans les images fixes. Nous avons utilisé dans notre évaluation les collections TRECVid 2013 pour les vidéos et Pascal VOC 2012 pour les images fixes. Contrairement à ce qui a été observé dans des travaux antérieurs concernant les images, pour le cas des vidéos, mais sans recourir à des ressources externes, l'approche de fusion des deux détecteurs donne de meilleurs résultats par rapport à la méthode basée sur l'apprentissage direct d'un modèle par paire de concepts. De plus, les approches de fusion sont également beaucoup plus faciles à mettre en œuvre et ne nécessitent ni un re-apprentissage ni de réglages et optimisations de paramètres. En ce qui concerne les images fixes, les deux types d'approches ont obtenu globalement des résultats comparables. Cependant chacune des deux méthodes s'est montrée plus efficace pour la détection d'une catégorie spécifique de paires/triplets. Nos différentes méthodes décrites dans ce travail restent applicables à n'importe quel type de documents multimédia.

Chapitre 7

Conclusion et perspectives

7.1 Conclusion

Les travaux présentés dans cette thèse s'inscrivent dans le cadre d'utilisation du contexte pour l'amélioration de la performance des systèmes d'indexation de documents image et vidéo. Nous avons vu que la notion de *contexte* a été largement utilisée dans l'état de l'art et que sa définition dépend spécifiquement du problème abordé. Nous rappelons que dans notre problématique, le but est d'améliorer la performance de détecteurs de concepts cibles. Les détecteurs sont censés se baser sur un apprentissage supervisé et renvoyer un score de détection, qui est interprété comme la probabilité que le document considéré contienne le concept cible. Nous avons montré que les concepts n'apparaissent pas dans les documents indépendamment les uns des autres. Par conséquent, tenter de détecter des concepts sans tenir compte des relations les reliant à d'autres concepts s'avère une idée naïve. Les sources de contexte peuvent être par exemple les métadonnées, les relations entre concepts, les relations spatiales et/ou temporelles entre les objets apparaissant dans les documents, etc.

Nous avons retenu dans le cadre de cette thèse deux catégories de contexte : 1) Le contexte sémantique et 2) Le contexte temporel pour le cas des vidéos. Même avec cette sélection, beaucoup de sources de contexte peuvent être considérées et plusieurs types d'approches sont possibles pour les exploiter. Nous avons visé des approches génériques pouvant être appliquées à n'importe quels systèmes d'indexation sans devoir apporter de changements conséquents dans les système d'origine. Par conséquent, nous avons considéré comme source de contexte sémantique différentes relations entre les concepts. En ce qui concerne le cas de vidéo, nous avons retenu les relations entre les plans d'une même vidéo comme source de contexte temporel. D'autres part, nous nous sommes intéressés au problème de la détection simultanée de groupes de concepts dans les images et vidéos

Cela nous a mené à un ensemble de contributions que nous répertorions en quatre catégories. Nous allons les résumer dans ce qui suit.

7.1.1 Contributions dans cette thèse

1. Contributions pour l'utilisation du contexte sémantique : nous avons présentés trois approches différentes :
   (a) La première approche intitulée "Ré-ordonnancement sémantique basée sur une ontologie", exploite les relations inter-concepts qui peuvent être issues d'une hiérarchie d'une ontologie ou d'une liste de relations de type "c_1 implique c_2" pour reclasser les résultats d'une première classification en modifiant le score de détection calculé dans la première étape. Même si cette méthode a amélioré un système ayant déjà une bonne performance, le gain n'était pas spectaculaire ;
   (b) La deuxième contribution : "reclassement sémantique par regroupement", modélise le contexte sémantique en regroupant les exemples selon leurs contenus sémantiques issus d'une étape préalable d'apprentissage. Le calcul de nouveaux scores de détection se fait via une combinaison de valeurs calculées après une étape de regroupement (clustering). Même si cette méthode a pu améliorer un bon système d'indexation, elle contient plusieurs paramètres et reste sujette au problème du sur-apprentissage ;
   (c) La troisième contribution : "rétroaction conceptuelle", consiste à construire un nouveau descripteur de haut niveau à partir des scores obtenus via une première étape de classification. Cette approche met à jour les scores de détection du concept cible en considérant une nouvelle étape d'apprentissage qui utilise le descripteur conceptuel généré. Nous avons proposé ensuite, une extension de cette approche, que nous avons appelée "rétroaction conceptuelle étendue". Cette dernière suit les mêmes étapes que la rétroaction conceptuelle classique, mais diffère sur la méthode de construction du descripteur conceptuel. Nous avons élaboré deux stratégies. Dans la première, nous avons effectué une sélection explicite des concepts à prendre en compte en ne gardant que ceux qui ont un lien sémantique avec le concept cible. Nous nous sommes basés pour effectuer cette sélection sur la hiérarchie d'une ontologie. Dans la deuxième stratégie, nous avons proposé de pondérer les dimensions conceptuelles, en donnant une importance aux concepts qui est relative à leurs corrélations par rapport au concept cible. Cette dernière stratégie s'est avérée plus efficace et a donné de meilleurs résultats que l'approche classique de rétraction. Finalement, nous avons montré que toutes les approches de rétroaction présentées peuvent être lancées d'une manière itérative. Une étude des résultats des itérations a montré que les résultats de la rétroaction classique sont améliorés au fur des itérations, mais ces derniers commencent à chuter à partir de l'itération 2, suite à une saturation due au cumul de bruit. La rétroaction étendue ne tire pas un avantage de l'approche itérative, et la stratégie de pondération des dimensions conceptuelles arrive en une seule itération à des résultats comparables à ceux obtenus dans la deuxième itération par la rétroaction classique itérative ;

2. Contributions pour l'utilisation du contexte temporel : nous avons présenté et comparé deux méthodes exploitant le contexte temporel pour l'indexation des vidéos par détection de concepts :
   (a) La première approche : "Re-scoring temporel", est une méthode de ré-ordonnancement qui consiste à modifier les soties d'un détecteur d'un concept en exploitant les scores de sa détection dans les plans voisins d'une même vidéo. Cette méthode qui utilise des informations sémantiques a montré son efficacité en améliorant plusieurs systèmes de référence en atteignant des gains relatifs sur la MAP inversement proportionnels à la performance des systèmes initiaux ;
   (b) La deuxième approche : "Descripteurs temporels", incorpore le contexte temporel dans une étape autre que le post-traitement des sorties de classification. En l'occurrence, celle de l'extraction des descripteurs. Cette approche combine les descripteurs extraits des plans voisins, en appliquant une fusion précoce de descripteurs appartenant à une fenêtre temporelle centrée sur le plan en question et considère donc un certain nombre de plans qui le précèdent et d'autres qui le suivent. Chacun de ces segments temporels concaténés est pondéré par un poids qui lui est attribué, de manière à donner plus d'importance aux plans qui sont plus proches du plan courant ; le poids de ce dernier sera le plus important. Malgré une optimisation simplifiée, cette méthode a amélioré un système de référence, mais elle s'est montrée moins efficace que le re-scoring temporel. Cependant, nous prévoyons de meilleures performances des descripteurs temporels avec une optimisation de la taille de la fenêtre temporelle et des poids des différents segments temporels considérés, surtout si cette optimisation est locale (pour chaque concept séparément) ;
3. Contributions pour l'utilisation conjointe des contextes sémantique et temporel : Nous avons présenté dans ce chapitre deux approches combinant les deux contextes sémantique et temporel pour l'indexation des vidéos par détection de concepts ;
   (a) La première proposition : "Ré-ordonnancement à deux couches", exploite les deux types de contexte séparément dans deux couches. Dans la première couche, les scores d'une première classification sont modifiés en utilisant des informations contextuelles (sémantiques et/ou temporelles). Dans la seconde couche, les résultats obtenus dans la première couche sont exploités pour calculer de nouveaux scores de détection des concepts. Les résultats ont montré que pour cette stratégie, il vaut mieux appliquer le re-soring temporel sur les résultats d'un re-scoring sémantique. Bien que cette méthode améliore les résultats, elle reste vulnérable à la propagation des erreurs et est sujette au problème du sur-apprentissage et aussi au problème de normalisation des scores ;
   (b) La deuxième contribution : "Rétroaction conceptuo-temporellle", consiste à exploiter de l'information temporelle et de l'incorporer dans l'approche de rétroaction conceptuelle présentée précédemment.

Cette proposition consiste à construire des descripteurs conceptuels à partir des scores de détection de concepts, et d'appliquer pour chaque plan, une fusion précoce de ces descripteurs de haut niveau appartenant à une fenêtre temporelle centrée sur le plan en question. Ces nouveaux descripteurs conceptuo-temporels sont ensuite utilisés par un apprenant de la même manière qu'un descripteur de bas niveau. Contrairement à l'approche du re-scoring à deux couches, la rétroaction conceptuo-temporelle considère une phase d'apprentissage. Cela constitue un avantage majeur par rapport à l'approche à deux couches, car en utilisant un apprenant le problème de normalisation des scores est contourné. D'une part, le gain atteint par la rétroaction conceptuelle est plus important que celui obtenu par le re-scoring à deux couches qui souffre beaucoup plus de l'handicap de propagation des erreurs d'une étape à celle qui la suit ;

4. Contributions pour la détection simultanée de groupes de concepts : Nous avons comparé deux types d'approches pour détecter simultanément des groupes de concepts (multi-concepts) dans les images et vidéos. La détection de multi-concepts est une tâche qui s'inscrit bel et bien dans la problématique de l'utilisation du contexte. En effet, pour chaque groupe de concepts, un ou plusieurs concepts le formant peuvent être considérés comme source de contexte pour certains autres. La première contribution consiste en la construction d'un modèle par multi-concept en générant des annotations en se basant sur celles des concepts singuliers, et en utilisant une méthode d'apprentissage d'ensembles. La deuxième est basée sur la combinaison des sorties de détecteurs de concepts formant le groupe, et ne nécessite aucune étape supplémentaire d'apprentissage. Nous avons instancié et évalué les deux méthodes pour la détection de paires de concepts dans les vidéos, et les paires et les triplets de concepts dans les images fixes. Nous avons utilisé dans notre évaluation les collections TRECVid 2013 pour les vidéos et Pascal VOC 2012 pour les images fixes. Contrairement à ce qui a été observé dans des travaux antérieurs concernant les images, pour le cas des vidéos, mais sans recourir à des ressources externes, l'approche de fusion des deux détecteurs donne de meilleurs résultats par rapport à la méthode basée sur l'apprentissage direct d'un modèle par paire de concepts. De plus, les approches de fusion sont également beaucoup plus faciles à mettre en œuvre et ne nécessitent ni un re-apprentissage ni de réglages et optimisations de paramètres. En ce qui concerne les images fixes, les deux types d'approches ont obtenu globalement des résultats comparables. Cependant chacune des deux méthodes s'est montrée plus efficace pour la détection d'une catégorie spécifique de paires/triplets. Nos différentes méthodes décrites dans ce travail restent applicables à n'importe quel type de documents multimédia ;

La plupart de nos contributions ont donné de bons résultats et ont atteint une performance comparable aux meilleurs systèmes évalués dans la compagne TRECVid/VOC. À titre comparatif, nous recommandons les approches de la rétroaction conceptuelle ainsi que son extension qui consiste à pondérer les di-

mensions conceptuelles, pour la détection des concepts visuels dans les documents multimédia. Pour le cas des vidéos, nous conseillons d'utiliser la méthode du re-scoring temporel, pour sa simplicité et surtout son efficacité. Les autres approches utilisant le contexte temporel restent efficaces, mais présentent l'inconvénient du coût de leur implémentation. L'approche du ré-ordonnancement sémantique basée sur l'ontologie reste une bonne alternative quand on vise une méthode simple sans traitements lourds. D'autre part, pour détecter des multi-concepts, nous recommandons d'utiliser les méthodes basées sur la fusion quand on dispose de bons détecteurs de concepts singuliers. Générer un détecteur par multi-concept reste un bon choix en présence d'un corpus de données suffisamment annoté.

7.1.2 Points forts de la thèse et limitations

Poins forts : Parmi les points forts de cette thèse nous citons la généricité de toutes nos approches. En effet, nos contributions sont applicables sur n'importe quel système de détection de n'importe quel concept dans les images/vidéos, et qui suit le pipeline : "Extraction de descripteurs/Classification supervisée/Fusion", ou encore mieux : tous les systèmes d'indexation de documents multimédia renvoyant des scores de détection de concepts. En outre, toutes nos approches ont été évaluées étant appliquées sur un système de référence ayant déjà une bonne performance, et étant bien classé dans la tâche d'indexation sémantique de TRECVid. Cela et contrairement à plusieurs travaux de l'état de l'art, démontre la robustesse de nos approches qui ne sont pas efficaces uniquement sur des systèmes de faibles performances. D'autres part, l'évaluation de nos approches a été réalisée dans le contexte de la compagne TRECVid, qui est reconnue comme un standard par la communauté, et même si nous n'avons pas fait des soumissions officielles pour chacun de nos résultats, nous avons suivi le même protocole expérimental et nous avons utilisé les mêmes outils d'évaluation. Un autre point fort de cette thèse est la variété des types de contributions. En effet, nous avons proposé différentes approches s'inscrivant chacune dans un niveau particulier d'un système classique d'indexation d'images/vidéos : extraction de descripteurs, fusion et ré-ordonnancement, rétroaction, etc.

Limitations : Ce travail de thèse présente quelques limitations. Premièrement, même si les approches ont été évaluées sur les corpus de données de TRECVid, il est tout de même important de valider les résultats sur d'autres corpus de données. D'autres part, l'étude des résultats n'a pas été suffisamment approfondie, pour voir le comportement des contributions avec différentes catégories de concepts : Objets, Événements, Véhicules, Endroits, etc. Ceci s'explique par le fait que l'un des objectifs principaux de cette thèse était la proposition d'approches génériques, pouvant être appliquées sur n'importe quel système, pour la détection de n'importe quel concept.

7.2 Perspectives

Perspectives à court terme :

Nous aimerions valider les résultats obtenus en appliquant nos approches sur d'autres corpus de données, notamment ceux de VOC et ImageClef. D'autres part, nous sommes restés à un niveau général en considérant des approches génériques. Même si cette caractéristique s'avère un atout pour nos contributions, il est important d'étudier plus en détails les résultats et la variation du gain sur des catégories de concepts différentes. En effet, il serait judicieux de connaitre dans quelles mesures et pour quels types particuliers de concepts nos approches sont susceptibles d'être plus efficaces, et aussi, de savoir pourquoi elles ne se comportent pas bien pour les autres catégories de concepts.

Perspectives à moyen et long terme :

Nous avons considéré pour le contexte sémantique un seul type de relations : 1) Relation d'implication ou relation père-fils dans une hiérarchie d'ontologie et 2) Relation d'exclusion. Il serait intéressant de tester d'autres types de relations entre concepts. D'autre part, il y a beaucoup de choses intéressantes que nous aimerions réaliser dans le cadre de la problématique étudiée dans ce travail de thèse, on va citer quelques unes dans ce qui suit.

Utilisation de noyaux prenant en compte le contexte : Nous avons vu que l'utilisation d'un classificateur (dans l'approche de rétroaction conceptuelle) est beaucoup plus efficace que la combinaison a posteriori des détecteurs de concepts. Ceci est dû au fait qu'un tel classificateur considère les relations entre les concepts durant son apprentissage. Cela nous amène à renforcer notre hypothèse qui stipule que l'utilisation d'un noyau permettant d'intégrer les relations inter-concepts ou n'importe quelles autres informations contextuelles pourrait améliorer encore mieux les performances des systèmes d'indexation des images et vidéos. Les noyaux sur les graphes [Dri] ou les données structurées [Gär03] est une bonne piste pour commencer l'étude de cette idée.

Construction d'une ontologie de contexte : Nous avons supposé tout au long de ce travail, que l'on disposait de détecteurs d'un ensemble de concepts ; de plus, même si nous avons considéré dans une de nos approches une hiérarchie de concepts nous n'avons pas utilisé une ontologie spécifique à la notion de *contexte*. Il serait judicieux de construire une ontologie codant les différents contextes possibles dans lesquels chaque concept peut apparaitre. Cela permettrait une utilisation optimale des relations de co-occurrences entre les concepts. Ainsi, on pourrait non seulement détecter la présence d'un concept mais aussi spécifier plus de détails sur son occurrence. Par exemple, si on arrive à détecter un policier dans une image ou vidéo, il serait important et intéressant de spécifier la situation dans laquelle il apparait : sortant de son véhicule, entrain de poursuivre un dealer, avec

son arme, etc. Disposer d'une ontologie qui encode les différents contextes possibles dans lesquels le concept "policier" pourrait apparaitre aiderait à atteindre cet objectif. En plus de la bonne détection de concepts, cela permettrait d'identifier les contextes dans lesquels ils apparaissent et aussi une analyse sémantique approfondie des images et vidéos, et améliorera sans doute la performance des moteurs de recherche sémantique et les rendra capables de répondre efficacement à des requêtes complexes.

Utilisation du contexte explicite : Nous avons présenté dans le premier chapitre différents types de contexte pour lesquels plusieurs sources d'information se présentent. Ces types de contexte se sont révélés cruciaux pour la détection de concepts dans les images et vidéos. Cependant, chacun de ces différents contextes est approprié à un une catégorie particulière ou une liste spécifique de concepts. Par exemple, le contexte spatial semble être plus utile pour la détection d'objets dans les images ou les concepts qui apparaissent souvent dans des endroits particuliers d'une image/vidéo, comme le concept "ciel" qui apparait généralement en haut de l'image ou de la scène vidéo. De même, le contexte temporel peut s'avérer indispensable pour la détection des concepts de types "évènement" ou les concepts dont l'apparition dure souvent longtemps. D'autre part, l'exploitation de certaines informations contextuelles peut dégrader la performance de détection de certains concepts. De ce fait, il devient important de savoir quels types de contexte sont utiles (respectivement inutiles) pour la détection des concepts. Ces d'informations peuvent être considérées par exemple lors de la construction d'une ontologie de contexte comme présenté ci-avant, afin de l'enrichir et de rendre possible de choisir les contextes à exploiter en fonction des concepts à détecter.

Identification des concepts *alarmes* pour corriger les erreurs de classification : Jusqu'à présent, nous avons fait en sorte de trouver les contextes pouvant aider à améliorer la détection des concepts visuels. Le problème c'est que les détecteurs de concepts ont un taux d'erreurs qui n'est pas nul. On n'est donc pas en mesure de savoir quand le détecteur se trompe, puisqu'il est possible d'avoir une fausse réponse de la part d'un bon détecteur. Cette remarque reste donc valable dans le cas d'un détecteur avec une performance basse. Pour remédier à cela, une idée consiste à étudier la corrélation entre les mauvaises décisions du détecteur en question et celles des détecteurs d'autres concepts, pour identifier les concepts qui sont susceptibles de signaler les mauvaises réponses de notre détecteur. Par exemple, quand le détecteur 1 trouve un "véhicule" dans une image, et que les détecteur 2 et 3 trouvent dans la même image une "voiture" et un "nuage", respectivement, alors la réponse du détecteur 2 est à considérer*fausse*, parce qu'il s'agit d'une *erreur* de sa part, et le véhicule identifié est un "avion" ou une "hélicoptère". On peut appeler le concept "nuage" un concept *alarme* puisqu'il signale la présence d'une erreur de détection.

Traitement du problème de normalisation de scores : Dans le cadre de cette thèse, nous avons fait un effort pour normaliser les distributions des scores

de nos classificateurs. Mais la tâche s'est avérée difficile à cause du problème des classes déséquilibrées. Nous sommes intéressés d'aborder cette problématique plus en détails pour voir ensuite ce que donnerait nos différentes approches décrites dans le cadre de cette thèse, avec des scores bien calibrés. Nous pensons que l'utilisation de scores calibrés pourrait aider à atteindre de meilleures performances.

Utilisation du contexte pour la détection simultanée de plusieurs concepts : Considérer des informations contextuelles (e.g. Le contexte, des ontologies, des annotations supplémentaires, récolte des échantillons positifs, etc.) pour la détection de groupes de concepts dans le documents multimédia fait partie de nos objectifs. Les résultats encourageants obtenus dans le cadre de cette problématique nous encouragent pour étendre notre travail et appliquer nos approches contextuelles ainsi que d'autres plus élaborées et adaptées au cas des multi-concepts. Par ailleurs, nous sommes également intéressés par l'extension de nos expérimentation et la considération de groupes de concepts plus riches au lieu de seulement deux/trois concepts, et étudier les résultats plus profondément. Un autre objectif intéressant serait d'appliquer nos méthodes sur les mêmes corpus utilisés dans l'état de l'art pour pouvoir confirmer ou infirmer certaines hypothèses, surtout celles qui sont en conflit avec les résultats obtenus dans nos expérimentations, et faire aussi une comparaison plus détaillée.

Exploitation du contexte via l'apprentissage profond : Motivés par le succès de l'apprentissage profond, surtout les réseaux de neurones à convolution (CNN), nous nous intéressons à l'exploiter en considérant le contexte pour l'indexation et la recherche sémantique des images et vidéos. Les CNNs sont connus pour ne pas nécessiter de pré-traitements lourds et par leur adaptation et leur capacité à réaliser un apprentissage efficace à partir des pixels des images. Étant génériques, nos différentes contributions dans le cadre de cette thèse peuvent être exploitées par ce genre d'approches. En effet, en utilisant un système d'indexation basé sur les CNNs on peut exploiter ses sorties correspondant aux scores de détection d'un ensemble de concepts via n'importe laquelle de nos approches de re-scoring (sémantique ou temporel) et/ou de rétroaction. De plus, la méthode "Descripteurs temporels" dont les résultats n'ont pas été convaincants par rapport aux autres à cause de la difficulté d'optimisation des paramètres peut bénéficier encore mieux des avantages des CNNs. Une manière de le faire est de considérer en entrée des CNNs non seulement un seul plan (ou image) mais une fenêtre temporelle (un ensemble de plans successifs), et là, les CNNs sont mieux adaptés et plus capables de réaliser un apprentissage plus efficace que la méthode suivie dans le cadre de cette thèse pour exploiter les descripteurs temporels.

Annexe A

Annexe A

A.1 Le concepts TRECVid évalués

Voir le tableau A.1.

A.2 Les concepts TRECVid utilisés (ceux évalués et non)

Here we present the concepts of the TRECVid 2010 and 2011, which will also be evaluated in 2012. In the TRECVid 2010 there are 130 concepts, whereas in 2011 the task was enlarged, however we succeeded to annotate 346 concepts over 500. The concepts that were used in TRECVid 2010 are the first 130 concepts in the following list.

1. Actor 2. Adult 3. Airplane 4. Airplane_Flying 5. Anchorperson 6. Animal 7. Asian_People 8. Athlete 9. Basketball 10. Beach 11. Beards 12. Bicycles 13. Bicycling 14. Birds 15. Boat_Ship 16. Boy 17. Bridges 18. Building 19. Bus 20. Canoe 21. Car 22. Car_Racing 23. Cats 24. Celebrity_Entertainment 25. Chair 26. Charts 27. Cheering 28. Cityscape 29. Classroom 30. Computer_Or_Television_Screens 31. Computers 32. Conference_Room 33. Construction_Vehicles 34. Corporate-Leader 35. Court 36. Cows 37. Crowd 38. Dancing 39. Dark-skinned_People 40. Daytime_Outdoor 41. Demonstration_Or_Protest 42. Desert 43. Dogs 44. Doorway 45. Driver 46. Eaters 47. Emergency_Vehicles 48. Entertainment 49. Explosion_Fire 50. Face 51. Female_Person 52. Female-Human-Face-Closeup 53. Flowers 54. Girl 55. Golf 56. Government-Leader 57. Greeting 58. Ground_Vehicles 59. Hand 60. Handshaking 61. Harbors 62. Helicopter_Hovering 63. Highway 64. Horse 65. Hospital 66. House_Of_Worship 67. Indoor 68. Indoor_Sports_Venue 69. Industrial_Setting 70. Infants 71. Instrumental_Musician 72. Kitchen 73. Laboratory 74. Landscape 75. Male_Person 76. Maps 77. Meeting 78. Military 79. Military_Base 80. Motorcycle 81. Mountain 82. Natural-Disaster 83. News_Studio 84. Nighttime 85. Office 86. Old_People 87. Outdoor 88. Overlaid_Text 89. People_Marching 90. Person 91. Plant 92. Police_Private_Security_Personnel 93. Politicians 94. Politics 95. Press_Conference 96. Prisoner 97. Reporters 98. Road 99. Roadway_Junction 100. Running 101. Scene_Text 102. Science_Technology 103. Scientists 104. Shopping_Mall 105. Singing 106. Single_Person 107. Sit-

ting_Down 108. Sky 109. Snow 110. Soccer_Player 111. Sports 112. Stadium 113. Streets 114. Suburban 115. Swimming 116. Teenagers 117. Telephones 118. Tennis 119. Tent 120. Throwing 121. Trees 122. Truck 123. Two_People 124. US_Flags 125. Vegetation 126. Vehicle 127. Walking 128. Walking_Running 129. Waterscape_Waterfront 130. Weather 131. 3_Or_More_People 132. Adult_Female_Human 133. Adult_Male_Human 134. Advocate 135. Airplane_Landing 136. Airplane_Takeoff 137. Airport_Or_Airfield 138. Amateur_Video 139. Anger 140. Animal_Pens_And_Cages 141. Animation_Cartoon 142. Apartment_Complex 143. Apartments 144. Armed_Person 145. Armored_Vehicles 146. Arthropod 147. Attached_Body_Parts 148. Baby 149. Background_Static 150. Bar_Pub 151. Baseball 152. Black_Frame 153. Blank_Frame 154. Body_Parts 155. Bomber_Bombing 156. Boredom 157. Car_Crash 158. Carnivore 159. Cattle 160. Caucasians 161. Cell_Phones 162. Cetacean 163. Child 164. Church 165. Cigar_Boats 166. City 167. Civilian_Person 168. Clearing 169. Clouds 170. Colin_Powell 171. Commentator_Or_Studio_Expert 172. Commercial_Advertisement 173. Conference_Buildings 174. Construction_Site 175. Construction_Worker 176. Crane_Vehicle 177. Crustacean 178. Cul-de-Sac 179. Dining_Room 180. Disgust 181. Dolphin 182. Domesticated_Animal 183. Door_Opening 184. Dresses 185. Dresses_Of_Women 186. Earthquake 187. Election_Campaign 188. Election_Campaign_Address 189. Election_Campaign_Convention 190. Election_Campaign_Debate 191. Election_Campaign_Greeting 192. Eukaryotic_Organism 193. Event 194. Exiting_A_Vehicle 195. Exiting_Car 196. Factory 197. Factory_Worker 198. Fear 199. Female_Anchor 200. Female_Human_Face 201. Female_News_Subject 202. Female_Reporter 203. Fields 204. Fighter_Combat 205. Fight-Physical 206. Fire_Truck 207. First_Lady 208. Flags 209. Flood 210. Food 211. Football 212. Forest 213. Free_Standing_Structures 214. Freighter 215. Furniture 216. George_Bush 217. Glasses 218. Golf_Player 219. Graphic 220. Ground_Combat 221. Guard 222. Gun 223. Gun_Shot 224. Gym 225. Head_And_Shoulder 226. Helicopters 227. Herbivore 228. High_Security_Facility 229. Hill 230. Hispanic_Person 231. Hockey 232. Human_Young_Adult 233. Indian_Person 234. Insect 235. Insurgents 236. Invertebrate 237. Islands 238. Japanese 239. John_Kerry 240. Joy 241. Junk_Frame 242. Korean 243. Lakes 244. Legs 245. Machine_Guns 246. Male_Anchor 247. Male_Human_Face 248. Male_News_Subject 249. Male_Reporter 250. Male-Human-Face-Closeup 251. Mammal 252. Man_Made_Thing 253. Man_Wearing_A_Suit 254. Military_Aircraft 255. Military_Airplane 256. Military_Buildings 257. Military_Personnel 258. Military_Vehicle 259. Minivan 260. Moonlight 261. Mosques 262. Muslims 263. Network_Logo 264. News 265. Oceans 266. Office_Building 267. Officers 268. Oil_Drilling_Site 269. Pan_Zoom_Static 270. Pavilions 271. Person_Drops_An_Object 272. Pickup_Truck 273. Police 274. Police_Car 275. Police_Truck 276. Primate 277. Processing_Plant 278. Professional_Video 279. Quadruped 280. Raft 281. Religious_Building 282. Religious_Figures 283. Rescue_Helicopter 284. Rescue_Vehicle 285. Researcher 286. Residential_Buildings 287. Rifles 288. River 289. Road_Block 290. Road_Overpass 291. Rocky_Ground 292. Room 293. Rowboat 294. Rpg 295. Ruminant 296. Sadness 297. Sailing_Ship 298. School 299. Sea_Mammal 300. Security_Checkpoint 301.

Single_Person_Female 302. Single_Person_Male 303. Skating 304. Ski 305. Skier 306. Skyscraper 307. Soccer 308. Sofa 309. Soldiers 310. Speaker_At_Podium 311. Speaking 312. Speaking_To_Camera 313. Sports_Car 314. Standing 315. Still_Image 316. Street_Battle 317. Studio_With_Anchorperson 318. Suits 319. Sun 320. Sunglasses 321. Sunny 322. Surprise 323. Swimming_Pools 324. Synagogue 325. Synthetic_Images 326. Table 327. Talking 328. Taxi_Cab 329. Text 330. Text_Labeling_People 331. Text_On_Artificial_Background 332. Throw_Ball 333. Tower 334. Traffic 335. Underwater 336. Urban_Park 337. Urban_Scenes 338. Valleys 339. Van 340. Vertebrate 341. Violent_Action 342. Weapons 343. Whale 344. Wild_Animal 345. Windows 346. Yasser_Arafat

A.3 Les concepts de ImageCLEF

1. Partylife 2. Family_Friends 3. Beach_Holidays 4. Building_Sights 5. Snow 6. Citylife 7. Landscape_Nature 8. Sports 9. Desert 10. Spring 11. Summer 12. Autumn 13. Winter 14. Indoor 15. Outdoor 16. Plants 17. Flowers 18. Trees 19. Sky 20. Clouds 21. Water 22. Lake 23. River 24. Sea 25. Mountains 26. Day 27. Night 28. Sunny 29. Sunset_Sunrise 30. Still_Life 31. Macro 32. Portrait 33. Overexposed 34. Underexposed 35. Neutral_Illumination 36. Motion_Blur 37. Out_of_focus 38. Partly_Blurred 39. No_Blur 40. Single_Person 41. Small_Group 42. Big_Group 43. No_Persons 44. Animals 45. Food 46. Vehicle 47. Aesthetic_Impression 48. Overall_Quality 49. Fancy 50. Architecture 51. Street 52. Church 53. Bridge 54. Park_Garden 55. Rain 56. Toy 57. MusicalInstrument 58. Shadow 59. bodypart 60. Travel 61. Work 62. Birthday 63. Visual_Arts 64. Graffiti 65. Painting 66. artificial 67. natural 68. technical 69. abstract 70. boring 71. cute 72. dog 73. cat 74. bird 75. horse 76. fish 77. insect 78. car 79. bicycle 80. ship 81. train 82. airplane 83. skateboard 84. female 85. male 86. Baby 87. Child 88. Teenager 89. Adult 90. old_person 91. happy 92. funny 93. euphoric 94. active 95. scary 96. unpleasant 97. melancholic 98. inactive 99. calm

Year	Concepts
2009	1.Classroom 2.Chair 3.Infant 4.Traffic-intersection 5.Doorway 6.Airplane_flying 7.Person-playing-a-musical-instrument 8.Bus 9.Person-playing-soccer 10.Cityscape 11.Person-riding-a-bicycle 12.Telephone 13.Person-eating 14.Demonstration_Or_Protest 15.Hand 16.People-dancing 17.Nighttime 18.Boat_Ship 19.Female-human-face-closeup 20.Singing*
2010	1.Airplane_Flying 2.Animal 3.Asian_People 4.Bicycling 5.Boat_Ship 6.Bus 7.Car_Racing 8.Cheering 9.Cityscape 10.Classroom 11.Dancing 12.Dark-skinned_People 13.Demonstration_Or_Protest 14.Doorway 15.Explosion_Fire 16.Female-Human-Face-Closeup 17.Flowers 18.Ground_Vehicles 19.Hand 20.Mountain 21.Nighttime 22.Old_People 23.Running 24.Singing 25.Sitting_Down 26.Swimming 27.Telephones 28.Throwing 29.Vehicle 30.Walking
2011	1.Adult 2.Anchorperson 3.Beach 4.Car 5.Charts 6.Cheering 7.Dancing 8.Demonstration_Or_Protest 9.Doorway 10.Explosion_Fire 11.Face 12.Female_Person 13.Female-Human-Face-Closeup 14.Flowers 15.Hand 16.Indoor 17.Male_Person 18.Mountain 19.News_Studio 20.Nighttime 21.Old_People 22.Overlaid_Text 23.People_Marching 24.Reporters 25.Running 26.Scene_Text 27.Singing 28.Sitting_down 29.Sky 30.Sports 31.Streets 32.Two_People 33.Walking* 34.Walking_Running 35.Door_Opening 36.Event 37.Female_Human_Face 38.Flags 39.Head_And_Shoulder 40.Male_Human_Face 41.News 42.Quadruped 43.Skating 44.Speaking 45.Speaking_To_Camera 46.Studio_With_Anchorperson 47.Table 48.Text 49.Traffic 50.Urban_Scenes
2012	1.Airplane 2.Airplane_Flying 3.Basketball 4.Bicycling 5.Boat_Ship 6.Boy 7.Bridges 8.Chair 9.Computers 10.Female_Person 11.Girl 12.Government-Leader 13.Greeting 14.Highway 15.Instrumental_Musician 16.Kitchen 17.Landscape 18.Male_Person 19.Meeting 20.Motorcycle 21.Nighttime 22.Office 23.Press_Conference 24.Roadway_Junction 25.Scene_Text 26.Singing 27.Sitting_Down 28.Stadium 29.Teenagers 30.Throwing 31.Walking_Running 32.Apartments 33.Baby 34.Civilian_Person 35.Clearing 36.Fields 37.Forest 38.George_Bush 39.Glasses 40.Hill 41.Lakes 42.Man_Wearing_A_Suit 43.Military_Airplane 44.Oceans 45.Skier 46.Soldiers
2013	1.Airplane 2.Anchorperson 3.Animal 4.Beach 5.Boat_Ship 6.Boy 7.Bridges 8.Bus 9.Chair 10.Computers 11.Dancing 12.Explosion_Fire 13.Female-Human-Face-Closeup 14.Flowers 15.Girl 16.Government_Leader 17.Hand 18.Instrumental_Musician 19.Kitchen 20.Motorcycle 21.News_Studio 22.Old_People 23.People_Marching 24.Running 25.Singing 26.Sitting_Down 27.Telephones 28.Throwing 29.Baby 30.Door_Opening 31.Fields 32.Flags 33.Forest 34.George_Bush 35.Military_Airplane 36.Quadruped 37.Skating 38.Studio_With_Anchorperson

Table A.1 – Les concepts évalués dans les dernières collections TRECVid.'*' indique les concepts communs entre les différentes collections.

Annexe B

Annexe B

B.1 Descripteurs de video(Genérés par les partenaires IRIM)

Nine IRIM participants (CEA-LIST, ETIS/LIP6, EURECOM, GIPSA, INRIA, LABRI, LIF, LIG, and LSIS) provided a total of 48 descriptors, including variants of same descriptors. Here we present these descriptors :

CEALIST/tlep : texture local edge pattern ([CC03]) + color histogram $\rightsquigarrow$ 576 dimensions.

ETIS/global_<feature>[<type>]x<size> : (concatenated) histogram features ([GCP11]), where :

<**feature**> is chosen among lab and qw :

lab : CIE L*a*b* colors

qw : quaternionic wavelets (3 scales, 3 orientations)

<**type**> can be

nothing : histogram computed on the whole image

m1x3 : histogram for 3 vertical parts

m2x2 : histogram on 4 image parts

<**size**> is the dictionary size, sometimes different from the final feature vector dimension.

For instance, with <type>=m1x3 and <size>=32, the final feature vector has $3 \times 32 = 96$ dimensions.

EUR/sm462 : the Saliency Moments (SM) feature ([RM11]), is a holistic descriptor that embeds locally-parsed information, namely the shape of the salient region, in a holistic representation of the scene, structurally similar to [OT01]. First, the saliency information is extracted at different resolutions using a spectral, light-weight algorithm. The signals obtained are then sampled directly in the frequency domain, using a set of Gabor wavelets. Each of these samples, called "Saliency Components", is then interpreted as a probability distribution : the components are divided into sub-windows

and the first three moments are extracted, namely mean, standard deviation and skewness. The resulting signature vector is a 462-dimensional descriptor that we use as input for traditional support vector machines and combine then with the contributions of the other visual features.

GIPSA/AudioSpectro[N]-b28 : Spectral profile in 28 bands on a Mel scale, N : normalized $\rightsquigarrow$ 28 dimensions.

INRIA/dense_sift_<k> : Bag of SIFT computed by INRIA with k-bin histograms $\rightsquigarrow$ k dimensions with k = 128, 256, 512, 1024, 2048 and 4096.

LEAR/sift_bow4096 : Bag Of SIFT Words vectors with dictionary size equal to 4096.

LABRI/faceTracks : OpenCV+median temporal filtering, assembled in tracks, projected on keyframe with temporal and spatial weighting and quantized on image divided in 16 $\times$ 16 blocks $\rightsquigarrow$ 256 dimensions.

LIF/percepts_<x>_<y>_1_15 : 15 mid-level concepts detection scores computed on x $\times$ y grid blocks in each key frames with (x,y) = (20,13), (16,6), (5,3), (2,2) and (1,1), $\rightsquigarrow$ 15 $\times$ x $\times$ y dimensions.

KIT/faces KIT contributed by proposing descriptors/predictions at the face level.

LIG/h3d64 : normalized RGB Histogram 4 $\times$ 4 $\times$ 4 $\rightsquigarrow$ 64 dimensions.

LIG/gab40 : normalized Gabor transform, 8 orientations $\times$ 5 scales, $\rightsquigarrow$ 40 dimensions.

LIG/hg104 : early fusion (concatenation) of h3d64 and gab40 $\rightsquigarrow$ 104 dimensions.

LIG/opp_sift_<method>[_unc]_1000 : bag of word, opponent sift, generated using [vdSGS08a] software. $\rightsquigarrow$ 1000 dimensions (384 dimensions per detected point before clustering ; clustering on 535117 points coming from 1000 randomly chosen images). <**method**> method is related to the way by which SIFT points are selected : **har** corresponds to a filtering via a Harris-Laplace detector and **dense** corresponds to a dense sampling ; the versions with **_unc** correspond to the same with fuzziness introduced in the histogram computation.

LIG/stip_<method>_<k> : bag of word, STIP local descriptor, generated using [Lap05] software, <**method**> may be either histograms of oriented (spatial) gradient (**hog**) or histograms of optical flow (**hof**), $\rightsquigarrow$ k dimensions with k = 256 or 1000.

LIG_concepts : detection scores on the 346 TRECVid 2011 SIN concepts using the best available fusion with the other descriptors, $\rightsquigarrow$ 346 dimensions.

LISTIC/SURF_retinaMasking_<k>_cross : SURF based bag of words (BOW) with k = 1024 or 4096 dimensions using a real-time retina model ([BCDH10]). We consider 40 frames around each sub-shot keyframe. An automatic salient blobs segmentation is applied on each frame and a dense grid is considered only within these regions. SURF descriptors are captured within each frame blobs and are cumulated along the 40 frames. This allows the BOW

of the subshot keyframe to be defined globally. Descriptors are extracted from the retinal foveal vision model (Parvocellular pathway). It allows light and noise robustness and enhanced SURF description. The retinal motion channel (Magnocellular pathway) is used to perform the automatic blobs segmentation. This channel allows transient blobs to be detected during the 40 frames. Such transient blobs are related to salient detailed areas during the retina model transient state (during the 20 first frames). Its also corresponds to moving areas at the retina's stable state (during the last 20 frames). Such segmentation allows spatio-temporal low level saliency areas to be detected. For BOW training, vocabulary learning is performed with Kmeans on 1008 subshots taken from 2011a and 2011b keyframes lists using 6 622 198 points.

LSIS/mlhmslbp_spyr_<k> : Three kinds of parameters based on a Multi-Level Histogram of Multi-Scale features including spatial pyramid technique (MLHMS) ([PG10]). In each parameters extraction method, the pictures were considered as gray-scale pictures. The two first kinds of parameters are based on local binary pattern (LBP). A two levels pyramid was used with the level being the entire picture and the second level being a half in the horizontal direction and a forth in the vertical direction respectively a third and a sixth for the second kind of parameters). Moreover, an overlapping of half of the level-direction size is used. 4 levels of scaling were also computed for the LBP parameters, from 1 to 4 pixels blocks. The resulting parameter vectors are then L2-clamp normed. For the third kind of parameters, we used second order Local Derivative Pattern (LDP). We used the same kind of level, scaling and spatial pyramid than for the two preceding parameters. The dimensions of the resulting vectors are respectively 10240 and 26624 for the MLHMS-LBP parameters, and 106496 for the MLHMS-LDP parameters. For practical reasons, we were only able to use the MLHMS-LBP descriptor with 10240 dimensions.

Annexe C

Annexe C

C.1 Liste de publications

Journaux internationaux :

1. Abdelkader Hamadi, Philippe Mulhem, Georges Quénot. Extended conceptual feedback for semantic multimedia indexing.Multimedia Tools and Applications. Publised online, apr 2014.

Conférences Internationales/ Workshops internaionaux, avec comité de lecture :

1. Abdelkader Hamadi, Georges Quénot, Philippe Mulhem. Annotation of still images by multiple visual concepts. 12th Internationnal Workshop on Content-Based Multimedia Indexing, Klagenfurt, Austria, jun 2014.
2. Abdelkader Hamadi, Philippe Mulhem, Georges Quénot.Infrequent concept pairs detection in multimedia documents. ACM International Conference on Multimedia Retrieval. To appear. :2081-2084, Glasgow, Scotland, apr 2014.
3. Abdelkader Hamadi, Georges Quénot, Philippe Mulhem. Clustering based re-scoring for semantic indexing of multimedia documents. Content-Based Multimedia Indexing (CBMI), 2013 11th International Workshop on, :41-46, Veszprém, Hungary, June 2013.
4. Abdelkader Hamadi, Georges Quénot, Philippe Mulhem. Conceptual Feedback for Semantic Multimedia Indexing. Content-Based Multimedia Indexing (CBMI), 2013 11th International Workshop on, 53-58, Veszprém, Hungary, June 2013.
5. Abdelkader Hamadi, Georges Quénot, Philippe Mulhem. Two-layers reranking approach based on contextual information for visual concepts detection in videos. CBMI, :1-6, jun 2012.

Workshops internaionaux sans comité de lecture :

1. Nicolas Ballas, Benjamin Labbé, Hervé Le Borgne, Philippe Gosselin, Miriam Redi, Bernard Merialdo, Rémi Vieux, Boris Mansencal, Jenny Benois-Pineau, Stéphane Ayache, Abdelkader Hamadi, Bahjat Safadi, Thi-Thu-Thuy Vuong, Dong Han, Nadia Derbas, Georges Quénot, Boyang Gao,

Chao Zhu, Yuxing tang, Emmanuel Dellandrea, Charles-Edmond Bichot, Liming Chen, Alexandre Benoît, Patrick Lambert, Tiberius Strat. IRIM at TRECVID 2013 : Semantic Indexing and Instance Search. Proc. TRECVID Workshop, Gaithersburg, MD, USA, nov 2013.

2. Bahjat Safadi, Nadia Derbas, Abdelkader Hamadi, Franck Thollard, Georges Quénot, Jonathan Delhumeau, Hervé Jégou, Tobias Gehrig, Hazim Kemal Ekenel, Rainer Stifelhagen. Quaero at TRECVID 2012 : Semantic Indexing. Proc. TRECVID Workshop, Gaithersburg, MD, USA, nov 2012.
3. Nicolas Ballas, Benjamin Labbé, Aymen Shabou, Hervé Le Borgne, Philippe Gosselin, Miriam Redi, Bernard Merialdo, Hervé Jégou, Jonathan Delhumeau, Rémi Vieux, Boris Mansencal, Jenny Benois-Pineau, Stéphane Ayache, Abdelkader Hamadi, Bahjat Safadi, Franck Thollard, Nadia Derbas, Georges Quénot, Hervé Bredin, Matthieu Cord, Boyang Gao, Chao Zhu, Yuxing tang, Emmanuel Dellandrea, Charles-Edmond Bichot, Liming Chen, Alexandre Benoît, Patrick Lambert, Tiberius Strat, Joseph Razik, Sébastion Paris, Hervé Glotin, Tran Ngoc Trung, Dijana Petrovska Delacrétaz, Gérard Chollet, Andrei Stoian, Michel Crucianu. IRIM at TRECVID 2012 : Semantic Indexing and Instance Search. Proc. TRECVID Workshop, Gaithersburg, MD, USA, nov 2012.
4. Abdelkader Hamadi, Bahjat Safadi, Thi-Thu-Thuy Vuong, Dong Han, Nadia Derbas, Philippe Mulhem, Georges Quénot. Quaero at TRECVID 2013 : Semantic Indexing and Instance Search. Proc. TRECVID Workshop, Gaithersburg, MD, USA, nov 2013.
5. Bahjat Safadi, Nadia Derbas, Abdelkader Hamadi, Franck Thollard, Georges Quénot, Hervé Jégou, Tobias Gehrig, Hazim Kemal Ekenel, Rainer Stifelhagen. Quaero at TRECVID 2011 : Semantic Indexing and Multimedi Event Detection. TREC Video Retrieval Evaluation workshop, Gaithersburg, MD USA, dec 2011.

Conférences internationales :

1. Abdelkader Hamadi, Philippe Mulhem, Georges Quénot. Annotation de vidéos par paires rares de concepts. CORIA 2014, Mar 2014.
2. Abdelkader Hamadi. Reclassement sémantique pour l'indexation de documents multimédia. CORIA2013 - 8e Rencontres Jeunes Chercheurs en Recherche d'Information (RJCRI), Neuchâtel, Suisse, Avril 2013.

Bibliographie

[Abb10] Sunitha Abburu. Context ontology construction for cricket video. *International Journal on Computer Science & Engineering*, 2 :2593 – 2597, December 2010.

[AHdVdJ08] Robin Aly, Djoerd Hiemstra, Arjen de Vries, and Franciska de Jong. A probabilistic ranking framework using unobservable binary events for video search. In *7th ACM International Conference on Content-based Image and Video Retrieval, CIVR 2008*, pages 349–358, New York, NY, USA, July 2008. ACM.

[Ald97] John Aldrich. R. a. fisher and the making of maximum likelihood 1912-1922. *Statistical Science*, 12(3) :162–176, 1997.

[AOS07] Yusuf Aytar, O. Bilal Orhan, and Mubarak Shah. Improving semantic concept detection and retrieval using contextual estimates. In *ICME*, pages 536–539, 2007.

[AQ08a] Stéphane Ayache and Georges Quénot. Video corpus annotation using active learning. In Craig Macdonald, Iadh Ounis, Vassilis Plachouras, Ian Ruthven, and RyenW. White, editors, *Advances in Information Retrieval*, volume 4956 of *Lecture Notes in Computer Science*, pages 187–198. Springer Berlin Heidelberg, 2008.

[AQ08b] Stéphane Ayache and Georges Quénot. Video Corpus Annotation using Active Learning. In *European Conference on Information Retrieval (ECIR)*, pages 187–198, Glasgow, Scotland, Mar. 2008.

[AQG07] Stéphane Ayache, Georges Quénot, and Jrme Gensel. Classifier fusion for svm-based multimedia semantic indexing. In Giambattista Amati, Claudio Carpineto, and Giovanni Romano, editors, *Advances in Information Retrieval, 29th European Conference on IR Research, ECIR 2007, Rome, Italy, April 2-5, 2007, Proceedings*, volume 4425 of *Lecture Notes in Computer Science*, pages 494–504. Springer, 2007.

[ASH96] Hisashi Aoki, Shigeyoshi Shimotsuji, and Osamu Hori. A shot classification method to select effective key-frames for video browsing. In Philippe Aigrain, Wendy Hall, Thomas D. C. Little, and V. Michael Bove Jr., editors, *ACM Multimedia*, pages 1–10. ACM Press, 1996.

[ASS01] Erin L. Allwein, Robert E. Schapire, and Yoram Singer. Reducing multiclass to binary : A unifying approach for margin classifiers. *J. Mach. Learn. Res.*, 1 :113–141, September 2001.

[Aus68] D. P Ausubel. *Educational Psychology : A Cognitive View.* New York : Holt, Rinehart and Winston, 1968.

[Aus00] D. P Ausubel. *The Acquisition and Retention of Knowledge : a Cognitive View.* Dordrect ; Boston : Kluwer Academic Publishers, 2000.

[AVFRP84] Jr. Arthur V. Forman, Patricia J. Rowland, and Wade G. Pemberton. Contextual analysis of tactical scenes. In *Proc. SPIE 0485, Applications of Artificial Intelligence I, 189*, June 1984.

[Aya07] Stéphane Ayache. *Indexation de documents vidéos par concepts et par fusion de caractristiques audio, image et texte.* PhD thesis, INPG, 2007.

[Bar04] M. Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5(8) :617–629, 2004.

[BCDH10] A. Benoit, A. Caplier, B. Durette, and J. Herault. Using human visual system modeling for bio-inspired low level image processing. *Comput. Vis. Image Underst.*, 114(7) :758–773, July 2010.

[Ben09] Rachid Benmokhtar. *Fusion multi-niveaux pour l'indexation et la recherche multimédia par le contenu sémantique.* PhD thesis, École Nationale Supérieure des Télécommunications (ENST- Télécom Paris)- Institut Eurécom, Sophia Antipolis., 2009.

[BFG+96] Jeffrey R. Bach, Charles Fuller, Amarnath Gupta, Arun Hampapur, Bradley Horowitz, Rich Humphrey, Ramesh Jain, and Chiao-Fe Shu. Virage image search engine : An open framework for image management. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 76–87, 1996.

[BGV92] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA, 1992. ACM.

[BKS03] F. I. Bashir, A. A. Khokhar, and D. Schonfeld. Segmented trajectory based indexing and retrieval of video data. In *in proc. IEEE Int. Conf. Image Processing*, volume 9, pages 623–626, 2003.

[BKS07] F. I. Bashir, A. A. Khokhar, and D. Schonfeld. Real-time motion trajectory-based indexing and retrieval of video sequences. *IEEE Transactions on Multimedia*, page 9, 2007.

[BL93] T.O. Binford and T.S Levitt. Quasi-invariants : Theory and exploitation. In *In Proceedings of DARPA Image Understanding Workshop*, pages 819–829, 1993.

[BL97] Avrim L. Blum and Pat Langley. Selection of relevant features and examples in machine learning. *ARTIFICIAL INTELLIGENCE*, 97 :245–271, 1997.

[BL02] Erwin M. Bakker and Michael S. Lew. Semantic video retrieval using audio analysis. In Michael S. Lew, Nicu Sebe, and John P. Eakins, editors, *CIVR*, volume 2383 of *Lecture Notes in Computer Science*, pages 271–277. Springer, 2002.

[BLCM03] I. Bloch, J.-P. Le Cadre, and H. Matre. Approches probabilistes et statistiques. In I. Bloch, editor, *Fusion d'informations en traitement du signal et des images*, Trait IC2, chapter 6, pages 87–118. Herms, 2003.

[BLS+12] Nicolas Ballas, Benjamin Labbé, Aymen Shabou, Hervé Le Borgne, Philippe Gosselin, Miriam Redi, Bernard Merialdo, Hervé Jégou, Jonathan Delhumeau, Rémi Vieux, Boris Mansencal, Jenny Benois-Pineau, Stéphane Ayache, Abdelkader Hamadi, Bahjat Safadi, Franck Thollard, Nadia Derbas, Georges Quénot, Hervé Bredin, Matthieu Cord, Boyang Gao, Chao Zhu, Yuxing tang, Emmanuel Dellandrea, Charles-Edmond Bichot, Liming Chen, Alexandre Benoît, Patrick Lambert, Tiberius Strat, Joseph Razik, Sébastion Paris, Hervé Glotin, Tran Ngoc Trung, Dijana Petrovska Delacrétaz, Gérard Chollet, Andrei Stoian, and Michel Crucianu. IRIM at TRECVID 2012 : Semantic Indexing and Instance Search. In *Proc. TRECVID Workshop*, Gaithersburg, MD, USA, 2012.

[BLSB04] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9) :1757–1771, 2004.

[BMG90] A.C. Bovik, M.Clark, and W.S. Geisler. Multichannl texture analysis using localized spatial filters. *IEEE Trans. Pattern Anal.Machine Intel.*, 12 :55–73, 1990.

[BMM99] R. Brunelli, O. Mich, and C. M. Modena. A Survey on the Automatic Indexing of Video Data,, ,. *Journal of Visual Communication and Image Representation*, 10(2) :78–112, June 1999.

[BP95] Aaron F. Bobick and Claudio S. Pinhanez. Using approximate models as source of contextual information for vision processing. In *Proceedings of the IEEE Workshop on Context-Based Vision (CBVIS '95)*, 1995.

[BR96] John S. Boreczky and Lawrence A. Rowe. Comparison of video shot boundary detection techniques. pages 170–179, 1996.

[Bre94] L. Breiman. Bagging predictors. Technical Report 421, Technical report,, University of California Berkeley, 1994.

[Bré99] Patrick Brézillon. Context in problem solving : A survey. *Knowl. Eng. Rev.*, 14(1) :47–80, May 1999.

[BT97] F. Brémond and M. Thonnat. Issues in representing context illustrated by scene interpretation applications. In *Proc. of the International and Interdisciplinary Conference on Modeling and Using Context (CONTEX'97)*, February 1997.

[BZ95] Chellappa-R Lin C L Burlina, P and X Zhang. Context-based exploitation of aerial imagery. In *Proceedings of the IEEE Workshop on Context-Based Vision (CBVIS '95)*, 1995.

[CC00] W. Chen and S. F. Chang. Motion trajectory matching of video objects. *IS&T/ SPIE*, pages 544–553, 2000.

[CC03] Ya-Chun Cheng and Shu-Yuan Chen. Image classification using color, texture and regions. *Image Vision Comput.*, 21(9) :759–776, 2003.

[CCM+98] Shi-Fu Chang, W. Chen, Horace J. Meng, H. Sundaram, and D. Zhong. A fully automated content-based video search engine supporting spatiotemporal queries. *IEEE Trans. on CSVT*, 8, No. 5 :602–615, September 1998.

[CdFB04] P. Carbonetto, N. de Freitas, and K. Barnard. A Statistical model for general contextual object recognition, 2004.

[CHA05] Mbarek CHARHAD. *Modèles de Documents Vidéo basés sur le Formalisme des Graphes Conceptuels pour l'Indexation et la Recherche par le Contenu Sémantique.* PhD thesis, Université Joseph Fourier -Grenoble 1-, 2005.

[CHJ+06] S.-F. Chang, W. Hsu, W. Jiang, L. Kennedy, D. Xu, A. Yanagawa, and E. Zavesky. Columbia university trecvid-2006 video search and high-level feature extraction. in proc. trecvid workshop. In *In Proc. TRECVID Workshop*, 2006.

[CHV99] Olivier Chapelle, Patrick Haffner, and Vladimir Vapnik. Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks*, 10(5) :1055–1064, 1999.

[CLE05] A. Choksuriwong, H. Laurent, and B. Emile. A comparative study of objects invariant descriptor. *ORASIS*, 2005.

[CMM+11] Dan Claudiu Ciresan, Ueli Meier, Jonathan Masci, Luca Maria Gambardella, and Jürgen Schmidhuber. Flexible, high performance convolutional neural networks for image classification. In *IJCAI*, pages 1237–1242, 2011.

[CSS98] Marco La Cascia, Saratendu Sethi, and Stan Sclaroff. Combining textual and visual cues for content-based image retrieval on the world wide web. In *In IEEE Workshop on Content-based Access of Image and Video Libraries*, pages 24–28, 1998.

[CT91] T. M. Cover and J. A. Thomas. *Elements of information theory.* Wiley-Interscience, New York, NY, USA, 1991.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3) :273–297, 1995.

[DAM08] Hoiem Dere, Efros Alexei, A, and Hebert Martial. Putting objects in perspective. *International Journal of Computer Vision*, 80(1) :3–15, 2008.

[DDF+90] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE*, 41(6) :391–407, 1990.

[Del02] Frank Dellaert. The expectation maximization algorithm. Technical Report GIT-GVU-02-20, Technical report, College of Computing, Georgia Institute of Technology, 2002.

[Den99] Yining Deng. A region representation for image and video retrieval. *Ph.D thesis, University of California, Santa Barbara*, 1999.

[Dey01] Anind K. Dey. Understanding and using context. *Personal Ubiquitous Comput.*, 5(1) :4–7, January 2001.

[DHH+09] Santosh Kumar Divvala, Derek Hoiem, James Hays, Alexei A. Efros, and Martial Hebert. An empirical study of context in object detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[Dim99] Alexander Dimai. Rotation invariant texture description using general moment invariants and gabor filters. volume I, pages 391–398. In Proc. Of the 11th Scandinavian Conf. on Image Analysis, June 1999.

[Dir00] F. Dirfaux. Key frame selection to represent a video. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 2, pages 275–278 vol.2, 2000.

[dLHM09] Jan de Leeuw, Kurt Hornik, and Patrick Mair. Isotone optimization in r : Pool-adjacent-violators algorithm (pava) and active set methods. *Journal of Statistical Software*, 32(5), 2009.

[DMM98] Yining Deng, Student Member, and B. S. Manjunath. Netra-v : Toward an object-based video representation. *IEEE Transactions on Circuits and Systems for Video Technology*, 8 :616–627, 1998.

[Dow93] James Dowe. Content-based retrieval in multimedia imaging, 1993.

[DPS89] M. Desvignes, C. Porquet, and P. Spagnou. A tool for studying context in image sequences. In *Image Processing and its Applications, 1989., Third International Conference on*, pages 467–471, Jul 1989.

[DPS91] M. Desvignes, C. Porquet, and P. Spagnou. The use of context in image sequences interpretation. In *Proceedings of the 8e Congrès AFCET-RFIA*, 1991.

[Dri]

[ea93] W. Niblack et. al. The qbic project : Querying images by content using color, texture and shape. In *Proc. of the Conference Storage and Retrieval for Image and Video Databases (SPIE)*, volume 1908, pages 173–187, 1993.

[(ed00] Sylvie Jeannin (ed.). Iso/iec jtc1/sc29/wg11/n3321 :mpeg-7 visual part of experimentation model version 5.0. nordwijkerhout. March 2000.

[EVGW+10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2) :303–338, June 2010.

[Fcd01] Text of iso/iec 15 938-3 multimedia content description interface, part 3 : Visual. *Final Committee Draft, ISO/IEC/JTC1/SC29/WG11*, Doc. N4062, Mar 2001.

[FCF96] G Finlayson, S. Chatterjee, , and B. Funt. Color angular indexing. In *In proceeding of the European conference on Computer Vision, Cambridge, England*, pages 16–27, 1996.

[FDF94] G.D Finlayson, M.S Drew, and B. Funt. Color constancy : Generalised diagonal transforms suffice. *of the optical Society of America A*, 11(11) :3011-3019, November 1994.

[FE73] Martin A. Fischler and R.A. Elschlager. The representation and matching of pictorial structures. *Computers, IEEE Transactions on*, C-22(1) :67–92, Jan 1973.

[FEF+94] C. Faloutsos, W. Equitz, M. Flickner, W. Niblack, D. Petkovic, and R. Barber. Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3 :231–262, 1994.

[Fel98] Christiane Fellbaum, editor. *WordNet : An Electronic Lexical Database (Language, Speech, and Communication).* The MIT Press, illustrated edition edition, May 1998.

[FGL07] Jianping Fan, Yuli Gao, and Hangzai Luo. Hierarchical classification for automatic image annotation. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 111–118, New York, NY, USA, 2007. ACM.

[FHY09] Gerald Friedland, Hayley Hung, and Chuohao Yeo. Multi-modal speaker diarization of real-world meetings using compressed-domain video features. *Dans International Conference on Acoustics, Speech and Signal Processing*, 2009.

[FMK96] S. Abbasi F. Mokhtarian and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. In *Int. Workshop on Image DataBases and Multimedia Search, Amsterdam, The Netherlands*, pages 35–42, 1996.

[FNG01] S. Fine, J. Navratil, and R.A. Gopinath. A hybrid gmm/svm approach to speaker identification. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, volume 1, pages 417 –420 vol.1, 2001.

[FS97] Yoav Freund and Robert E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1) :119–139, 1997.

[FTTUG10] Ali Fakeri-Tabrizi, Sabrina Tollari, Nicolas Usunier, and Patrick Gallinari. Improving image annotation in imbalanced classification

problems with ranking svm. In Carol Peters, Barbara Caputo, Julio Gonzalo, GarethJ.F. Jones, Jayashree Kalpathy-Cramer, Henning Müller, and Theodora Tsikrika, editors, *Multilingual Information Access Evaluation II. Multimedia Experiments*, volume 6242 of *Lecture Notes in Computer Science*, pages 291–294. Springer Berlin Heidelberg, 2010.

[GA11] Mehmet Gönen and Ethem Alpaydin. Multiple kernel learning algorithms. *J. Mach. Learn. Res.*, 12 :2211–2268, July 2011.

[Gär03] Thomas Gärtner. A survey of kernels for structured data. *SIGKDD Explorations*, 5(1) :49–58, 2003.

[GB10] Carolina Galleguillos and Serge Belongie. Context based object categorization : A critical survey. *Computer Vision and Image Understanding (CVIU)*, 114 :712–722, 2010.

[GC08] Andrew C. Gallagher and Tsuhan Chen. Estimating age, gender, and identity using first name priors. In *CVPR*. IEEE Computer Society, 2008.

[GCP11] David Gorisse, Matthieu Cord, and Frédéric Precioso. Salsas : Sublinear active learning strategy with approximate k-nn search. *Pattern Recognition*, 44(10-11) :2343–2357, 2011.

[GFK05] Todor Ganchev, Nikos Fakotakis, and George Kokkinakis. Comparative evaluation of various mfcc implementations on the speaker verification task. In *10th International Conference on Speech and Computer (SPECOM-2005)*, pages 191–194, 2005.

[GLTT10] Magdalena Graczyk, Tadeusz Lasota, Bogdan Trawinski, and Krzysztof Trawinski. Comparison of bagging, boosting and stacking ensembles applied to real estate appraisal. In Ngoc Thanh Nguyen, Manh Thanh Le, and Jerzy Swiatek, editors, *ACIIDS (2)*, volume 5991 of *Lecture Notes in Computer Science*, pages 340–350. Springer, 2010.

[GNC+08] Andrew C. Gallagher, Carman G. Neustaedter, Liangliang Cao, Jiebo Luo, and Tsuhan Chen. Image annotation using personal calendars as context. In *MM '08 : Proceeding of the 16th ACM international conference on Multimedia*, pages 681–684, New York, NY, USA, 2008. ACM.

[Gru93] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. In *IN FORMAL ONTOLOGY IN CONCEPTUAL ANALYSIS AND KNOWLEDGE REPRESENTATION, KLUWER ACADEMIC PUBLISHERS, IN PRESS. SUBSTANTIAL REVISION OF PAPER PRESENTED AT THE INTERNATIONAL WORKSHOP ON FORMAL ONTOLOGY*. Kluwer Academic Publishers, 1993.

[HE08] James Hays and Alexei A. Efros. Im2gps : estimating geographic information from a single image. In *in IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[HEH07] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Recovering surface layout from an image. *International Journal of Computer Vision*, 75(1) :151–172, 2007.

[HEH11] Derek Hoiem, Alexei A. Efros, and Martial Hebert. Recovering occlusion boundaries from an image. *Int. J. Comput. Vision*, 91(3) :328–346, February 2011.

[HKm+97] J. Huang, S. Ravi Kumar, M. mitra, W.J Zhu, and R. Zabih. Image indexing using color correlograms. In *In Proceedings of the Conference on Computer Vision and pattern Recognition, Puerto Rico, USA*, pages 762–768, June 1997.

[HM99] G. M. Haley and B. S. Manjunath. Rotation invariant texture classification using a complete space-frequency model. *IEEE Trans. Image Processing*, 8 :255–269, Feb 1999.

[HM00] Riad Hammoud and Roger Mohr. A probabilistic framework of selecting effective key frames for video browsing and indexing. In *International workshop on Real-Time Image Sequence Analysis (RISA '00)*, pages 79–88, Oulu, Finlande, 2000.

[HMQ13] A. Hamadi, P. Mulhem, and G. Quenot. Conceptual feedback for semantic multimedia indexing. In *Content-Based Multimedia Indexing (CBMI), 2013 11th International Workshop on*, pages 53–58, 2013.

[HO00] A Hyvärinen and E Oja. Independent component analysis : algorithms and applications. *Neural Networks : The Official Journal of the International Neural Network Society*, 13(4-5) :411–430, jun 2000. PMID : 10946390.

[HQM12] A. Hamadi, G. Quenot, and P. Mulhem. Two-layers re-ranking approach based on contextual information for visual concepts detection in videos. In *Content-Based Multimedia Indexing (CBMI), 2012 10th International Workshop on*, pages 1–6, 2012.

[HQS00] N. Haering, R. J. Qian, and M. I. Sezan. A semantic event-detection apporach and its application to detecting hunts in wildlife video. *IEEE Trans. on CSVT*, 10, No. 6 :857–868, September 2000.

[HS06] Geoffrey E. Hinton and Ruslan Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313 :504–507, July 2006.

[Hu62] M. K. Hu. Visual pattern recognition by moment invariants. *IEEE Transactions Information Theory*, 8 :179–187, 1962.

[HW98] Alexander G. Hauptmann and Michael J. Witbrock. Story segmentation and detection of commercials in broadcast news video. In *Proceedings of Advances in Digital Libraries Conference*, pages 168–179, 1998.

[HyCC+04] A. Hauptmann, M. y. Chen, M. Christel, C. Huang, W. h. Lin, T. Ng, A. Velivelli, J. Yang, R. Yan, H. Yang, and H. D. Wactlar. Confounded expectations : Informedia at trecvid 2004. In *In Proc. of TRECVID*, 2004.

[IB95] S S. Intille and A F Bobick. Exploiting contextual information for tracking by using closed-worlds. In *Proceedings of the IEEE Workshop on Context-Based Vision (CBVIS '95)*, 1995.

[IMA09] Najlae Idrissi, José Martinez, and Driss Aboutajdine. Bridging the semantic gap for texture-based image retrieval and navigation. *Journal of Multimedia*, 4(5) :277–283, 2009.

[ISO01] ISO/IEC/JTC1/SC29/WG11. Information technology - multimedia content description interface - part 3 visual. *Doc. No. 4062, Singapore*, March 2001.

[IT95] Makoto Iwayama and Takenobu Tokunaga. Hierarchical bayesian clustering for automatic text classification. In *IJCAI*, pages 1322–1327. Morgan Kaufmann, 1995.

[JDH00] Tommi Jaakkola, Mark Diekhans, and David Haussler. A discriminative framework for detecting remote protein homologies. *Journal of Computational Biology*, 7(1-2) :95–114, 2000.

[JDM00] Anil K. Jain, Robert P. W. Duin, and Jianchang Mao. Statistical pattern recognition : A review. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 22(1) :4–37, 2000.

[JF91] A.K. Jain and F. Farrokhnia. Unsupervised texture segmentation using gabor filters. *Pattern Recognition*, 24, no. 12 :1167–1186, 1991.

[JH91] Christian Jutten and Jeanny Herault. Blind separation of sources, part 1 : An adaptive algorithm based on neuromimetic architecture. *Signal Process.*, 24(1) :1–10, August 1991.

[JH99] Tommi S. Jaakkola and David Haussler. Exploiting generative models in discriminative classifiers. In *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pages 487–493, Cambridge, MA, USA, 1999. MIT Press.

[JNR05] Anil K. Jain, Karthik Nandakumar, and Arun Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12) :2270–2285, 2005.

[JPD+11] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011. QUAERO.

[Kan01] Emmanuel Kant. *Critique de la raison pure(1787)*. PUF, 2001.

[Kap98] L M et al Kaplan. Fast texture database retrieval using extended fractal features in storage and retrieval for image and video databases vi (sethi, i k and jain, r c, eds). In *Proc SPIE*, volume 3312, pages 162–173, 1998.

[KC07] Lyndon S. Kennedy and Shih-Fu Chang. A reranking approach for context-based concept fusion in video indexing and retrieval. In *Proceedings of the 6th ACM international conference on Image*

and video retrieval, CIVR '07, pages 333–340, New York, NY, USA, 2007. ACM.

[KLY07] Ho Joon Kim, Joseph S. Lee, and Hyun Seung Yang. Human action recognition using a modified convolutional neural network. In Derong Liu, Shumin Fei, Zeng-Guang Hou, Huaguang Zhang, and Changyin Sun, editors, *ISNN (2)*, volume 4492 of *Lecture Notes in Computer Science*, pages 715–723. Springer, 2007.

[KS03] Hannes Kruppa and Bernt Schiele. Using local context to improve face detection. In *British Machine Vision Conference (BMVC)*, September 2003.

[KSH12] Alex Krizhevsky, I Sutskever, and G. E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS 2012)*, page 4, 2012.

[lal]

[Lap05] Ivan Laptev. On space-time interest points. *Int. J. Comput. Vision*, 64(2-3) :107–123, September 2005.

[LAW80] K. I. LAWS. Rapid texture identification. *July 29-August 1, 1980. (A81-39326 18-04) Bellingham, WA, Society of Photo-Optical Instrumentation Engineers*, pages 376–380, 1980.

[LJ06] Diane Larlus and Frédéric Jurie. Latent mixture vocabularies for object categorization. In Mike J. Chantler, Robert B. Fisher, and Emanuele Trucco, editors, *BMVC*, pages 959–968. British Machine Vision Association, 2006.

[LJH02] Wei-Hao Lin, Rong Jin, and Alexander G. Hauptmann. Meta-classification of multimedia classifiers. In *KDMCD*, pages 21–27, 2002.

[LJZ01] Lie Lu, Hao Jiang, and Hongjiang Zhang. A robust audio classification and segmentation method. In *ACM Multimedia*, pages 203–211, 2001.

[LKF10] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. Convolutional networks and applications in vision. In *ISCAS*, pages 253–256. IEEE, 2010.

[LNE10] Jean-François Lalonde, Srinivasa G. Narasimhan, and Alexei A. Efros. What do the sun and the sky tell us about the camera ? *International Journal on Computer Vision*, 88(1) :24–51, May 2010.

[LNSF02] Cheng-Lin Liu, Kazuki Nakashima, Hiroshi Sako, and Hiromichi Fujisawa. Handwritten digit recognition using state-of-the-art techniques. In *IWFHR*, pages 320–325. IEEE Computer Society, 2002.

[Low04] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2) :91–110, 2004.

[LP96] F. Liu and R W. Picard. Periodicity, directionality and randomness : Wold features for image modelling and retrieval. 18(7) :722–733, 1996.

[LP05] Fei-Fei Li and Pietro Perona. A bayesian hierarchical model for learning natural scene categories. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 524–531, Washington, DC, USA, 2005. IEEE Computer Society.

[LSC06] Lscom lexicon definitions and annotations version 1.0, dto challenge workshop on large scale concept ontology for multimedia. Technical report, Columbia University ADVENT Technical Report #217-2006-3, March 2006.

[LSCB99] C. S. Li, J. R. Smith, V. Castelli, and L. Bergman. Comparing texture feature sets for retrieving core images in petroleum aopplications. In *in Proc. SPIE, San Jose, CA*, volume 3656, pages 2–11, 1999.

[LSP06] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features : Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2169 – 2178, 2006.

[LSST+02] Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini, and Chris Watkins. Text classification using string kernels. *J. Mach. Learn. Res.*, 2 :419–444, March 2002.

[LSWS12] Xirong Li, C. G M Snoek, M. Worring, and A.W.M. Smeulders. Harvesting social images for bi-concept search. *Multimedia, IEEE Transactions on*, 14(4) :1091–1104, 2012.

[LTD+10] Yuanning Li, Yonghong Tian, Ling-Yu Duan, Jingjing Yang, Tiejun Huang, and Wen Gao. Sequence multi-labeling : A unified video annotation scheme with spatial and temporal context. *Multimedia, IEEE Transactions on*, 12(8) :814–828, Dec 2010.

[LWLZ07] Xirong Li, Dong Wang, Jianmin Li, and Bo Zhang. Video search in concept subspace : A text-like paradigm. In *In Proc. of CIVR*, 2007.

[LYT+08] Ce Liu, Jenny Yuen, Antonio Torralba, Josef Sivic, and William T. Freeman. Sift flow : Dense correspondence across different scenes. In *Proceedings of the 10th European Conference on Computer Vision : Part III*, ECCV '08, pages 28–42, Berlin, Heidelberg, 2008. Springer-Verlag.

[MA92] J.L Mundy and A.Zisserman. Geometric invariance in computer vision. *The MIT Press, Cambridge, MA, USA*, 11(11) :3011-3019, 1992.

[Ma97] Wei-Ying Ma. Netra : A toolbox for navigating large image databases. *Ph.D thesis, University of California, Santa Barbara*, 1997.

[MC09] Rui Min and H.D. Cheng. Effective image retrieval using dominant color descriptor and fuzzy support vector machine. *Pattern Recognition*, 42 :147–157, 2009.

[MEA10] S. Memar, M. Ektefa, and L.S. Affendey. Developing context model supporting spatial relations for semantic video retrieval. In *Information Retrieval Knowledge Management, (CAMP), 2010 International Conference on*, pages 40–43, March 2010.

[MGP03] Florent Monay and Daniel Gatica-Perez. On image auto-annotation with latent space models. In *ACM Multimedia*, pages 275–278, 2003.

[MGP04] Florent Monay and Daniel Gatica-Perez. Plsa-based image auto-annotation : constraining the latent space. In *ACM Multimedia*, pages 348–351, 2004.

[ML99] Eric Margolis and Stephen Laurence, editors. *Concepts : Core Readings*. MIT Press, 1999.

[MM96a] W. Y. MA and B. S. MANJUNATH. Texture features and learning similarity. *CVPR*, page 00 :425, 1996.

[MM96b] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18 :837–842, August 1996.

[Moo96] T. K. Moon. The expectation-maximization algorithm. *IEEE Signal Processing Magazine*, 13(6) :47–60, November 1996.

[MOVY01] B. S. Manjunath, J. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. In *Transactions on Circuits and Systems for Video Technology*, volume 11(6), pages 703–715, June 2001.

[MQ12] Sangwoo Moon and Hairong Qi. Hybrid dimensionality reduction method based on support vector machine and independent component analysis. *IEEE Trans. Neural Netw. Learning Syst.*, 23(5) :749–761, 2012.

[MR01] A. Mojsilovic and B. Rogowitz. Capturing image semantics with low-level descriptors. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 1, pages 18–21 vol.1, 2001.

[MSY05] S.P. Cho K.J. Lee M.H. Song, J.Lee and S.K. Yoo. Support vector machine based arrhythmia classification using reduced features. *International Journal of Control, Automation, and Systems*, 3(4) :571–579, 2005.

[MTF03] Kevin Murphy, Antonio Torralba, and William T. Freeman. Using the forest to see the trees : A graphical model relating features, objects, and scenes, 2003.

[MVB+11] Mauro Dalla Mura, Alberto Villa, Jon Atli Benediktsson, Jocelyn Chanussot, and Lorenzo Bruzzone. Classification of hyperspectral images by using extended morphological attribute profiles and independent component analysis. *IEEE Geosci. Remote Sensing Lett.*, 8(3) :542–546, 2011.

[MWK+05] Cynthia Matuszek, Michael Witbrock, Robert C. Kahlert, John Cabral, Dave Schneider, Purvesh Shah, and Doug Lenat. Searching

for common sense : Populating cyc from the web. In *IN PROCEEDINGS OF THE TWENTIETH NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, pages 1430–1435, 2005.

[MWNS00] B. S. Manjunath, P. Wu, S. Newsam, and H. Shin. A texture descriptor for browsing and image retrieval. *Int. Commun. J*, 16 :33–43, 2000.

[MZF93] J.L Mundy, A. Zisserman, and D. Forsyth. Application invariance in computer vision. *Lecture Notes in Computer Science, 825*, 1993.

[Nap04] Milind R. Naphade. On supervision and statistical learning for semantic multimedia analysis. *J. Vis. Comun. Image Represent.*, 15 :348–369, September 2004.

[NDC+11] J. Nagi, F. Ducatelle, G. A. Di Caro, D. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. In *International Joint Conference on Neural Networks IJCNN*, 2011.

[NG84] J. D. Novak and D. B Gowin. *Learning how to learn.* Cambridge University Press, New York, 1984.

[NH01a] M.R. Naphade and T.S. Huang. Detecting semantic concepts using context and audiovisual features. In *Detection and Recognition of Events in Video, 2001. Proceedings. IEEE Workshop on*, pages 92 –98, 2001.

[NH01b] M.R. Naphade and T.S. Huang. Recognizing high-level audio-visual concepts using context. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 3, pages 46 –49 vol.3, 2001.

[NJ01] Andrew Y. Ng and Michael I. Jordan. On discriminative vs. generative classifiers : A comparison of logistic regression and naive bayes, 2001.

[NKK+05] M-R. Naphade, L. Kennedy, J-R. Kender, S-F. Chang, J-R. Smith, P. Over, and A. Hauptmann. A light scale concept ontology for multimedia understanding for trecvid 2005 (lscom-lite). Technical report, IBM Research Technical Report, December 2005.

[NMC05] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In Luc De Raedt and Stefan Wrobel, editors, *ICML*, volume 119 of *ACM International Conference Proceeding Series*, pages 625–632. ACM, 2005.

[NN02] Srinivasa G. Narasimhan and Shree K. Nayar. Vision and the atmosphere. *International Journal of Computer Vision*, 48(3) :233–254, 2002.

[OAF+10] Paul Over, George Awad, Jonathan Fiscus, Brian Antonishek, Alan F. Smeaton, Wessel Kraaij, and Georges Quénot. TRECVid 2010 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms

and Metrics. In *Proceedings of TRECVid 2010*. NIST, USA, nov 2010.

[OAM+12] Paul Over, George Awad, Martial Michel, Jonathan Fiscus, Greg Sanders, Barbara Shaw, Wessel Kraaij, Alan F. Smeaton, and Georges Quénot. TRECVID 2012 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *Proceedings of TRECVID 2012*. NIST, USA, 2012.

[OT01] Aude Oliva and Antonio Torralba. Modeling the shape of the scene : A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42 :145–175, 2001.

[PBE+06] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, C. K. I. Williams, J. Zhang, and A. Zisserman. Dataset issues in object recognition. In *Toward Category-Level Object Recognition, volume 4170 of LNCS*, pages 29–48. Springer, 2006.

[PC14] P. H. O. Pinheiro and R. Collobert. Recurrent convolutional neural networks for scene labeling. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014.

[PD07a] Jose San Pedro and Sergio Dominguez. Network-aware identification of video clip fragments. In *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, CIVR '07, pages 317–324, New York, NY, USA, 2007. ACM.

[PD07b] Florent Perronnin and Christopher R. Dance. Fisher kernels on visual vocabularies for image categorization. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*. IEEE Computer Society, 2007.

[Pea01] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2(6) :559–572, 1901.

[PG10] Sébastien Paris and Hervé Glotin. Pyramidal multi-level features for the robot vision@icpr 2010 challenge. In *ICPR*, pages 2949–2952, 2010.

[Pla99] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

[PPS96] A. Pentland, R. W. Picard, and S. Sclaroff. Photobook : Content-based manipulation of image databases. *Int. J. Comput. Vision*, 18(3) :233–254, June 1996.

[Pro99] Jefferson Provost. Naïve-bayes vs. rule-learning in classification of email, 1999.

[PZ99] G. Pass and R. Zabih. Comparing images using joint histograms. *Multimedia Systems*, 3, 7 :234–240, 1999.

[PZM96] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *Proceedings of the fourth ACM international conference on Multimedia*, pages 65–73, 1996.

[QGWF10] Yu Qiu, Genliang Guan, Zhiyong Wang, and Dagan Feng. Improving news video annotation with semantic context. In *Digital Image Computing : Techniques and Applications (DICTA), 2010 International Conference on*, pages 214–219, Dec 2010.

[QHR+07] Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, Tao Mei, and Hong-Jiang Zhang. Correlative multi-label video annotation. In Rainer Lienhart, Anand R. Prasad, Alan Hanjalic, Sunghyun Choi, Brian P. Bailey, and Nicu Sebe, editors, *Proceedings of the 15th International Conference on Multimedia 2007, Augsburg, Germany, September 24-29, 2007*, pages 17–26. ACM, 2007.

[QHR+08] Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, and Hong-Jiang Zhang. Two-dimensional active learning for image classification. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0 :1–8, 2008.

[QHR+10] G.-J. Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, and Hong-Jiang Zhang. Image classification with kernelized spatial-context. *Multimedia, IEEE Transactions on*, 12(4) :278–287, June 2010.

[RC96] B. Reddy and B. Chatterji. An fft-based technique for translation, rotation, and scale-invariant image registration. *IEEE Trans. Image Proc.*, 5(8) :1266,1271, 1996.

[Res99] Philip Resnik. Semantic similarity in a taxonomy : An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11 :95–130, 1999.

[RM11] Miriam Redi and Bernard Merialdo. Saliency moments for image categorization. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, ICMR '11, pages 39 :1–39 :8, New York, NY, USA, 2011. ACM.

[RMBB89] R. Rada, H. Mili, E. Bicknell, and M. Blettner. Development and application of a metric on semantic nets. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(1) :17–30, Jan 1989.

[RNKH02] M. R Naphade, I. V. Kozintsev, and T. S. Huang. Factor graph framework for semantic video indexing. *IEEE Trans. Cir. and Sys. for Video Technol.*, 12(1) :40–52, January 2002.

[Ro98] Y. M. Ro. Matching pursuit : Contents featuring for image indexing. In *in Proc. SPIE*, volume 3527, pages 89–100, 1998.

[Rot95] C.A Rothwell. Object rocognition throught invariant indexing. *Oxford Science Publication*, 1995.

[RTL+07] Bryan C. Russell, Antonio Torralba, Ce Liu, Robert Fergus, and William T. Freeman. Object recognition by scene alignment. In *NIPS*, 2007.

[Rus02] J.C. Russ. Image processing handbook, fourth edition. *CRC Press, Inc*, 2002.

[RVG+07] Andrew Rabinovich, Andrea Vedaldi, Carolina Galleguillos, Eric Wiewiora, and Serge Belongie. Objects in context. In *ICCV*, pages 1–8, 2007.

[SAW94] B. Schilit, N. Adams, and R. Want. Context-aware computing applications. In *Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications*, WMCSA '94, pages 85–90, Washington, DC, USA, 1994. IEEE Computer Society.

[SC99] Mark Sanderson and Bruce Croft. Deriving concept hierarchies from text. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 206–213, New York, NY, USA, 1999. ACM.

[Sch90] Robert E. Schapire. The strength of weak learnability. *Mach. Learn.*, 5 :197–227, July 1990.

[Sch02] A. J. Schölkopf, B. et Smola. Support vector machines and kernel algorithms, the handbook of brain theory and neural networks, m. a. arbib (eds.). pages 1119–1125, 2002.

[SCS01] Ishwar K. Sethi, Ioana L. Coman, and Daniela Stan. Mining association rules between low-level image features and high-level concepts, 2001.

[SDH+11] Bahjat Safadi, Nadia Derbas, Abdelkader Hamadi, Franck Thollard, Georges Quénot, Hervé Jégou, Tobias Gehrig, Hazim Kemal Ekenel, and Rainer Stifelhagen. Quaero at TRECVid 2011 : Semantic Indexing and Multimedi Event Detection. In *TREC Video Retrieval Evaluation workshop*, Gaithersburg, MD USA, dec 2011. National Institute of Standards and Technology.

[SDH+12] Bahjat Safadi, Nadia Derbas, Abdelkader Hamadi, Franck Thollard, Georges Quénot, Jonathan Delhumeau, Hervé Jégou, Tobias Gehrig, Hazim Kemal Ekenel, and Rainer Stifelhagen. Quaero at TRECVID 2012 : Semantic Indexing. In *Proc. TRECVID Workshop*, Gaithersburg, MD, USA, 2012.

[SDHH98] Mehran Sahami, Susan Dumais, David Heckerman, and Eric Horvitz. A bayesian approach to filtering junk e-mail, 1998.

[SDI08] Gregory Shakhnarovich, Trevor Darrell, and Piotr Indyk. Nearest-neighbor methods in learning and vision. *IEEE Transactions on Neural Networks*, 19(2) :377, 2008.

[SF91] T.M. Strat and M.A. Fischler. Context-based vision : recognizing objects using information from both 2d and 3d imagery. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(10) :1050–1065, Oct 1991.

[SFW83] Gerard Salton, Edward A. Fox, and Harry Wu. Extended boolean information retrieval. *Commun. ACM*, 26(11) :1022–1036, November 1983.

[SGG+06] C. G. M. Snoek, J. C. Van Gemert, Th. Gevers, B. Huurnink, D. C. Koelma, M. Van Liempt, O . De Rooij, F. J. Seinstra, A. W. M. Smeulders, A. H. C. Thean, C. J. Veenman, and M. Worring. The mediamill trecvid 2006 semantic video search engine. In *In Proceedings of the 4th TRECVID Workshop*, 2006.

[SGN01] Nathan Smith, Mark Gales, and Mahesan Niranjan. Data-Dependent Kernels in SVM Classification of Speech Patterns. Technical report, Cambridge University Engineering Dept., 2001.

[SH96] D. Slater and G. Healey. The illumination-invariant recognition of 3d objects using color invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(2) :206-210, 1996.

[SHH+07] C. G.M. Snoek, B. Huurnink, L. Hollink, M. de Rijke, G. Schreiber, and M. Worring. Adding semantics to detectors for video retrieval. *Trans. Multi.*, 9(5) :975–986, August 2007.

[SIYM03] Robert Snelick, Mike Indovina, James Yen, and Alan Mink. Multimodal biometrics : Issues in design and testing. In *Proceedings of the 5th International Conference on Multimodal Interfaces*, ICMI '03, pages 68–72, New York, NY, USA, 2003. ACM.

[SM81] Edward E. Smith and Douglas L. Medin. *Categories and Concepts*. Harvard University Press, 1981.

[Smi97] John R. Smith. Integrated spatial and feature image system : Retrieval, analysis and compression. *Ph.D thesis, Columbia University*, 1997.

[SNN03] J. R. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *Proceedings of ICME - Volume 1*, pages 445–448, Washington, DC, USA, 2003. IEEE Computer Society.

[SO95] Markus A. Stricker and Markus Orengo. Similarity of Color Images. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 381–392, 1995.

[SOK06] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *MIR'06 : Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[SOK09] Alan F. Smeaton, Paul Over, and Wessel Kraaij. High-Level Feature Detection from Video in TRECVid : a 5-Year Retrospective of Achievements. In Ajay Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, Berlin, 2009.

[SQ10] Bahjat Safadi and Georges Quénot. Evaluations of multi-learners approaches for concepts indexing in video documents. In *RIAO*, Paris, France, Apr 2010.

[SQ11a] Bahjat Safadi and Georges Quénot. Re-ranking by local re-scoring for video indexing and retrieval. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management (CIKM)*, pages 2081–2084, Glasgow, United Kingdom, 2011.

[SQ11b] Bahjat Safadi and Georges Quénot. Re-ranking by Local Re-scoring for Video Indexing and Retrieval. In *CIKM 2011 : 20th ACM Conference on Information and Knowledge Management*, CIKM '11, pages 2081–2084, Glasgow, Scotland, Oct. 2011. ACM.

[SQ11c] Bahjat Safadi and Georges Quénot. Re-ranking for Multimedia Indexing and Retrieval. In *ECIR 2011 : 33rd European Conference on Information Retrieval*, pages 708–711, Dublin, Ireland, Apr. 2011. Springer.

[SQ13] Bahjat Safadi and Georges Quénot. Descriptor optimization for multimedia indexing and retrieval. In *Proc. of Content Based Multimedia Ingexing (CBMI) Workshop*, Veszprém, Hungary, June 2013.

[SRS06] S. Sonnenburg, G. Ratsch, and C. Schafer. A general and efficient multiple kernel learning algorithm. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 1273–1280. MIT Press, Cambridge, MA, 2006.

[SS94] M. Stricker and M. Swain. The capacity of color histogram indexing. In *In Proceedings of the Conference on Computer Vision and Pattern Recognition, Seattle, Washington USA*, 1994.

[SS01] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[SSPS09] Stefan Siersdorfer, Jose San Pedro, and Mark Sanderson. Automatic video tagging using content redundancy. In *Proceedings of the 32Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 395–402, New York, NY, USA, 2009. ACM.

[Ste07] Patrick Sterlin. Overfitting prevention with cross-validation. *Supervised Machine Learning Report*, 83, 2007.

[Str93] Thomas M. Strat. Employing contextual information in computer vision. In *In Proceedings of ARPA Image Understanding Workshop*, pages 217–229, 1993.

[SVH04] N. Seco, T. Veale, and J. Hayes. An intrinsic information content metric for semantic similarity in WordNet. *Proc. of ECAI*, 4 :1089 ?1090–1089 ?1090, 2004.

[SWHO11] Hui Shen, William J. Welch, and Jacqueline M. Hughes-Oliver. Efficient, adaptive cross-validation for tuning and comparing models, with application to drug discovery. *The Annals of Applied Statistics*, 5(4) :2668–2687, 12 2011.

[SWRC09] Jamie Shotton, John Winn, Carsten Rother, and Antonio Criminisi. Textonboost for image understanding : Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. *Int. J. Comput. Vision*, 81(1) :2–23, January 2009.

[SWS+00] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12) :1349–1380, December 2000.

[SWS05] Cees G. M. Snoek, Marcel Worring, and Arnold W. M. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 399–402, New York, NY, USA, 2005. ACM.

[SZ99] E. Sahouria and A. Zakhor. A trajectory based video indexing system for street surveillance. *IEEE Int. Conf. on Image Processing (ICIP)*, pages 24–28, 1999.

[SZ03] J. Sivic and A. Zisserman. Video Google : A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, October 2003.

[TAP12] Ruxandra Georgiana TAPU. *Segmentation et structuration des documents vidéos pour l'indexation.* PhD thesis, Telecom SUDPARIS et l'université Pierre et Marie Curie, 2012.

[TAS11] K. C. Tiwari, M. K. Arora, and D. Singh. An assessment of independent component analysis for detection of military targets from hyperspectral images. *Int. J. Applied Earth Observation and Geoinformation*, 13(5) :730–740, 2011.

[TCG08] Pierre Tirilly, Vincent Claveau, and Patrick Gros. Language modeling for bag-of-visual words image categorization. In Jiebo Luo, Ling Guan, Alan Hanjalic, Mohan S. Kankanhalli, and Ivan Lee, editors, *CIVR*, pages 249–258. ACM, 2008.

[TFMB04] A. Tremeau, C. Fernandez-Maloigne, and P. Bonton. Image numérique couleur, de l'acquisition au traitement. *Dunod*, 2004.

[TJ98] M. TUCERYAN and A. K. JAIN. Texture analysis. *In The Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, pages 207–248, 1998.

[TMY76] H. Tamura, S. Mori, and T. Yamawaki. Texture features corresponding to visual perception. 6(4) :460–473, 1976.

[Tor03] Antonio Torralba. Contextual priming for object detection. *Int. J. Comput. Vision*, 53(2) :169–191, July 2003.

[Tou72] S Toulmin. *Human Understanding. Volume 1 : The Collective Use and Evolution of Concepts.* Princeton, NJ : Princeton University Press., 1972.

[Tou78] Godfried T. Toussaint. The use of context in pattern recognition. *Pattern Recognition*, 10(3) :189–204, 1978.

[TSBB+12] Sabin Tiberius Strat, Alexandre Benoît, Hervé Bredin, Georges Quénot, and Patrick Lambert. Hierarchical late fusion for concept detection in videos. In *ECCV 2012, Workshop on Information Fusion in Computer Vision for Concept Recognition*, Firenze, Italy, Oct. 2012.

[Vap98] Vladimir N. Vapnik. *Statistical learning theory.* Wiley, 1 edition, sep 1998.

[vdSGS08a] K. van de Sande, T. Gevers, and C. Snoek. Evaluation of color descriptors for object and scene recognition. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1 –8, june 2008.

[vdSGS08b] Koen E.A. van de Sande, Theo Gevers, and Cees G.M. Snoek. A comparison of color features for visual concept classification. In *Proceedings of the 2008 international conference on Content-based image and video retrieval*, CIVR '08, pages 141–150, New York, NY, USA, 2008. ACM.

[VECR10] F. Vallet, S. Essid, J. Carrive, and G. Richard. Robust visual features for the multimodal identification of unregistered speakers in tv talk-shows. In *In Proc. International Conference on Image Processing (ICIP)*, 2010.

[VG01] Alexei Vinokourov and Mark Girolami. Document classification employing the fisher kernel derived from probabilistic hierarchic corpus representations. *Proceedings of ECIR01 23rd European Colloquium on Information Retrieval Research*, pages 24–40, 2001.

[vGMU96] L. van Gool, T. Moons, and D. Ungureanu. Affine/photometric invariants for planar intensity pattern. In *In Proceedings of the 4th European Conference on Computer Vision, Cambridge, england*, pages 642–651, 1996.

[VPGB02] Jaco Vermaak, Patrick Perez, Michel Gangnet, and Andrew Blake. Rapid summarisation and browsing of video sequences. In *In British Machine Vision Conference*, pages 424–433, 2002.

[Wal98] Lucien Wald. Data fusion : a conceptual approach for an efficient exploitation of remote sensing images. In Thierry Ranchin and Lucien Wald, editors, *Proceedings of the 2nd conference "Fusion of Earth data : merging point measurements, raster maps and remotely sensed images"*, pages 17–23. SEE/URISCA, 1998.

[WB06] Lior Wolf and Stanley Bileschi. A critical view of context. *International Journal of Computer Vision*, 69(2) :251–261, August 2006.

[WC] Ming-Fang Weng and Yung-Yu Chuang. Multi-cue fusion for semantic video indexing. In *In Proceeding of the 16th ACM international conference on Multimedia, MM '08, pages 71-80, New York, NY, USA, 2008. ACM. ACM ID : 1459370.*

[WC06] Jing Wang and Chein-I Chang. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. *IEEE T. Geoscience and Remote Sensing*, 44(6) :1586–1600, 2006.

[WC12] Ming-Fang Weng and Yung-Yu Chuang. Cross-domain multicue fusion for concept-based video indexing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(10) :1927–1941, Oct. 2012.

[WCCS04] Yi Wu, Edward Y. Chang, Kevin Chen-Chuan Chang, and John R. Smith. Optimal multimodal fusion for multimedia data analysis. In *Proceedings of the 12th annual ACM international conference on Multimedia*, MULTIMEDIA '04, pages 572–579, New York, NY, USA, 2004. ACM.

[WF09] Gang Wang and David A. Forsyth. Joint learning of visual attributes, object classes and visual saliency. In *ICCV'09*, pages 537–544, 2009.

[WHL+99] Huang Liu Wang, J. Huang, Z. Liu, Y. Wang, Y. Chen, and E. K. Wong. Integration of multimodal features for video scene classification based on hmm. In *In IEEE Workshop on Multimedia Signal Processing*, pages 53–58, 1999.

[Win01] Terry Winograd. Architectures for context. *Hum.-Comput. Interact.*, 16(2) :401–419, December 2001.

[WJN11] Xiao-Yong Wei, Yu-Gang Jiang, and Chong-Wah Ngo. Concept-driven multi-modality fusion for video search. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(1) :62–73, 2011.

[WKSS96] Howard D. Wactlar, Takeo Kanade, Michael A. Smith, and Scott M. Stevens. Intelligent access to digital video : Informedia project. *Computer*, 29(5) :46–52, May 1996.

[WM09] F. Wang and B. Merialdo. Eurecom at trecvid 2009 high-level feature extraction. In *TREC2009 notebook*, 16-17 Nov 2009.

[Wol92] David H. Wolpert. Stacked generalization. *Neural Networks*, 5 :241–259, 1992.

[WTS04] Y. Wu, B. L. Tseng, and J. R. Smith. Ontology-based multi-classification learning for video concept detection. volume 2, June 2004.

[XLWZ11] Wei Xia, Xuesong Liu, Bin Wang, and Liming Zhang. Independent component analysis for blind unmixing of hyperspectral imagery with additional constraints. *IEEE T. Geoscience and Remote Sensing*, 49(6-1) :2165–2179, 2011.

[YA06] Emine Yilmaz and Javed A. Aslam. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, CIKM '06, pages 102–111, New York, NY, USA, 2006. ACM.

[YH03] Rong Yan and Alexander G. Hauptmann. The combination limit in multimedia retrieval. In *In Proceedings of the eleventh ACM international conference on Multimedia (MULTIMEDIA Ó3*, pages 339–342, 2003.

[YH08a] Jun Yang and Alexander G. Hauptmann. (un)reliability of video concept detection. In *CIVR'08*, pages 85–94, 2008.

[YH08b] Yi-Hsuan Yang and Winston H. Hsu. Video search reranking via online ordinal reranking. In *ICME*, pages 285–288, 2008.

[YKY+13] Sun Yongqing, Sudo Kyoko, Taniguch Yukinobu, Li Haojie, Guan Yue, and Liui Lijuan. Trecvid 2013 semantic video concept detection by ntt-md-dut. In *In Proc. of TRECVID*, 2013.

[YLZ07] Jinhui Yuan, Jianmin Li, and Bo Zhang. Exploiting spatial context constraints for automatic image region annotation. In *Proceedings of the 15th International Conference on Multimedia*, MULTIMEDIA '07, pages 595–604, New York, NY, USA, 2007. ACM.

[ZE01] Bianca Zadrozny and Charles Elkan. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *In Proceedings of the Eighteenth International Conference on Machine Learning*, pages 609–616. Morgan Kaufmann, 2001.

[ZGZ02] W. Zeng, W. Gao, and D. Zhao. Video indexing by motion activity maps. In *Proc. of IEEE ICIP'02, Rochester, NY, Sept*, 2002.

[ZH00] Xiang S. Zhou and Thomas S. Huang. Cbir : from low-level features to high-level semantics, 2000.

[Zho12] Zhi-Hua Zhou. *Ensemble Methods : Foundations and Algorithms*. Chapman & Hall/CRC, 1st edition, 2012.

[ZK98] Tong Zhang and C.-C. Jay Kuo. Content-based classification and retrieval of audio. In *IN SPIES 43RD ANNUAL MEETING - CONFERENCE ON ADVANCED SIGNAL PROCESSING ALGORITHMS, ARCHITECTURES, AND IMPLEMENTATIONS VIII*, pages 432–443, 1998.

[ZL01] D. S. Zhang and G. J. Lu. Shape retrieval using fourier descriptors. In *Proc. Int. Conference on Multimedia and Distance Education (ICMADE-01), Fargo, ND, USA*, pages 1–9, June 2001.

[ZL03] D. Zhang and G. Lu. Evaluation of mpeg-7 shape descriptors against other shape descriptors. *ACM Journal of Multimedia Systems*, 9(1) :15–30, 2003.

[ZLS07] J. Zhang, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories : a comprehensive study. *International Journal of Computer Vision*, 73 :2007, 2007.

[ZRHM98] Yueting Zhuang, Yong Rui, Thomas S. Huang, and Sharad Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *ICIP (1)*, pages 866–870, 1998.

[ZRZH05] Ji Zhu, Saharon Rosset, Hui Zou, and Trevor Hastie. Multi-class adaboost. Technical report, 2005.

[ZWLX10] Yingbin Zheng, Renzhong Wei, Hong Lu, and Xiangyang Xue. Semantic video indexing by fusing explicit and implicit context spaces. In *Proceedings of the international conference on Multimedia*, MM '10, pages 967–970, New York, NY, USA, 2010. ACM.

[ZZ95] HongJiang Zhang and Di Zhong. Scheme for visual feature-based image indexing. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 36–46, 1995.

[ZZY+13] Hanwang Zhang, Zheng-Jun Zha, Yang Yang, Shuicheng Yan, Yue Gao, and Tat-Seng Chua. Attribute-augmented semantic hierarchy : Towards bridging semantic gap and intention gap in image retrieval. In *Proceedings of the 21st ACM International Conference on Multimedia*, MM '13, pages 33–42, New York, NY, USA, 2013. ACM.