Construction et Présentation des Vidéos Interactives

Riad Hammoud

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

N^o _ _ _ _ _ _ _ _ _ _

THESE

pour obtenir le grade de

DOCTEUR DE L'INPG

Spécialité : IMAGERIE, VISION ET ROBOTIQUE

préparée au laboratoire GRAVIR-IMAG et INRIA Rhône-Alpes
dans le cadre de l'école Doctorale
Mathématiques, sciences et technologie de l'information

présentée et soutenue publiquement

par

Riad HAMMOUD

le 27 février 2001

Construction et Présentation des Vidéos Interactives

Directeur de thèse : Roger MOHR

Jury

M. Augustin LUX	, President
Mme. Françoise PRETEUX	, Rapporteur
M. Theirry PUN	, Rapporteur
M. Liming CHEN	, Rapporteur
M. Roger MOHR	, Examinateur
Mme. Cordelia SCHMID	, Examinateur

à mes parents, mes soeurs et frères.

à Aba Saleh et Aba Hadi.

à Safi Dager.

Remerciements

Trois ans déjà ! Me voilà disant merci à ceux qui ont rendu cette aventure possible et tous ceux qui l'ont ornée d'échanges ou d'amitiés.

Je voudrais donc tout d'abord remercier Roger Mohr, mon directeur de thèse, pour m'avoir accueilli dans l'équipe MOVI. Je lui suis très reconnaissant d'avoir poursuivi mon encadrement lors de son séjour à Melbourne et durant sa direction du centre de recherche européen de XEROX à grenoble. Sa confiance est au commencement de tout.

Je tiens à remercier le centre de recherche d'ALCATEL d'avoir financer cette étude et aussi le chef du projet MOVI, Radu Horaud, pour sa proposition de financer les trois derniers mois (février, mars et avril) de mon séjour à l'INRIA RHÔNE ALPES. Également, je remercie l'équipe de Patrick Bouthemy pour sa collaboration, et aussi l'Institut National de l'Audiovisuel de nous avoir donné la permission d'utiliser leurs données vidéos.

Je remercie vivement les membres de mon jury pour l'honneur qu'ils m'ont fait. Merci à M. Augustin Lux d'avoir accepté de présider mon jury. Merci à MME. Françoise Prêteux et M. Thierry Pun, pour les remarques constructives qui ont permis d'améliorer ce mémoire. Merci à tous ceux qui ont rendu la lecture de ce mémoire et de mes publications plus agréable. Je voudrais en particulier cité Matthieu Personnaz, Roger Mohr, Augustin Lux, Bill Trrigs et Ragini Choudhury.

Je suis très reconnaissant aux gens qui ont consacré du temps pour faire un échange scientifique et technique. Je voudrais en particulier cité Christophe Biernacki, Partick Gros et Pascal Bertolino. Également, je salut tous les stagiaires qui ont travaillé sous ma direction : Donacien Guifokou, MME Leila Cammoun, Mathieu Rudant, Khassoum Wone, et Guillaum Durant.

Ma reconnaissance va encore vers tous les membres du projet MOVI, pour leur esprit d'équipe et leur cordialité, mais aussi plus largement aux membres des équipes IS2, PRIMA et OPERA.

Enfin, il me faut saluer tous les amis qui m'ont entouré pour leurs services et leurs encouragements jusqu'à ce dernier moment.

Sommaire

Table des figures

Chapitre 1

Introduction

Un projet d'entreprise ne peut tenir debout que sur trois pieds : un concept solide, un créateur déterminé et une préparation rigoureuse.

1.1 Contexte et motivation

Depuis l'apparition des standards de compression vidéo MPEG-1 et MPEG-2, l'évolution rapide des performances des processeurs et des technologies de stockage comme le CDROM et le DVD, et la forte présence du Web et la facilité d'y accéder via le réseau Internet, la vidéo numérique dispose aujourd'hui d'une technologie qui la rend accessible comme les autres données.

Les applications de la vidéo sont en constante augmentation : les services de la "vidéo à la demande", l'enseignement à distance, la vidéo surveillance, les bases de données vidéo de films et de journaux télévisés, la "vidéo en continu" sur le Web (*streaming*), "TV Any-Time", etc. De nouvelles fonctionnalités essentiellement liées à l'**interactivité** séduisent de plus en plus les grandes entreprises dans ce domaine d'applications. On peut ainsi dès à présent visionner sur le Web des vidéo panoramiques et obtenir automatiquement des informations supplémentaires à certains moments-clés d'un reportage dite de "**vidéo enrichie**".

1.1.1 Structures et hyperliens

Le document vidéo a une structure qui n'apparaît pas explicitement, à la différence d'un ouvrage avec ses chapitres et ses paragraphes. Permettre une manipulation avancée d'une vidéo demandera donc de mettre en évidence ces éléments.

Les premières entités à considérer sont l'équivalent des mots dans un texte: les objets élémentaires dans un film. Il nous faudra donc extraire de tels objets qui se confondent avec le fond; le chapitre 2 détaille cet aspect. L'équivalent de ce que pourrait être un

paragraphe serait le plan vidéo, et des structures de plus haut niveau existent comme les scènes et les séquences.

Une fois cette structure créée, des hyperliens sur les entités existantes jouent le même rôle que dans un document écrit; les liens peuvent typiquement être classés dans deux catégories :

- **Lien interne :** il lie des entités dans la vidéo, typiquement lorsqu'une relation chronologique et/ou sémantique existe entre deux ou plusieurs entités d'un document vidéo. Le même objet dans le plan suivant est un exemple (voir figure 1.1).
- **Lien externe :** il s'agit d'un lien classique permettant par exemple d'attacher une page web à un objet dans un plan. Un exemple d'application publicitaire serait ainsi d'attacher un descriptif technique à un véhicule visible dans une vidéo.

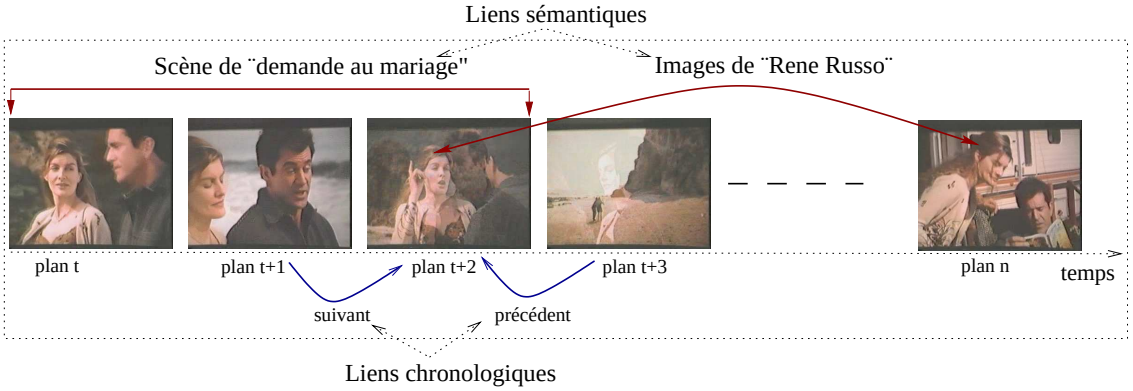


FIG. 1.1: *Exemple des hyperliens intra-film.*

1.1.2 Interactivité

Avec un magnétoscope classique, on peut visualiser un film vidéo, rebobiner pour observer un segment plusieurs fois, avec une vitesse rapide ou lente, sauter des segments, ou bien faire une pause. Ces capacités de contrôle de la vidéo rendent la vidéo interactive pour certains utilisateurs. Mais l'interaction ici est sévèrement limitée. Pour être vraiment interactive la vidéo doit offrir des niveaux d'interactions plus liés à son contenu; par exemple on peut imaginer l'option suivante : "*Indiquez l'acteur qui vous intéresse dans le film pour avoir plus d'information sur lui* ou bien "*Un clic sur un objet permettra de visualiser la scène suivante/précédente où il est présent*". Un exemple d'une vidéo interactive est illustré dans la figure 1.2.

Les propositions sur l'interactivité ont commencés à apparaître très récemment au début de l'année 1998. Plusieurs sociétés telles Intervu, Langages virtuels et Artsvidéo Interactive, ont proposé d'insérer de façon transparente aux vidéos une couche d'interactivité compatible avec les trois principaux formats de diffusion sur le Réseau (Realplayer, Quicktime et Windows Media Player). En 1999, deux nouveaux formats de vidéo enrichie

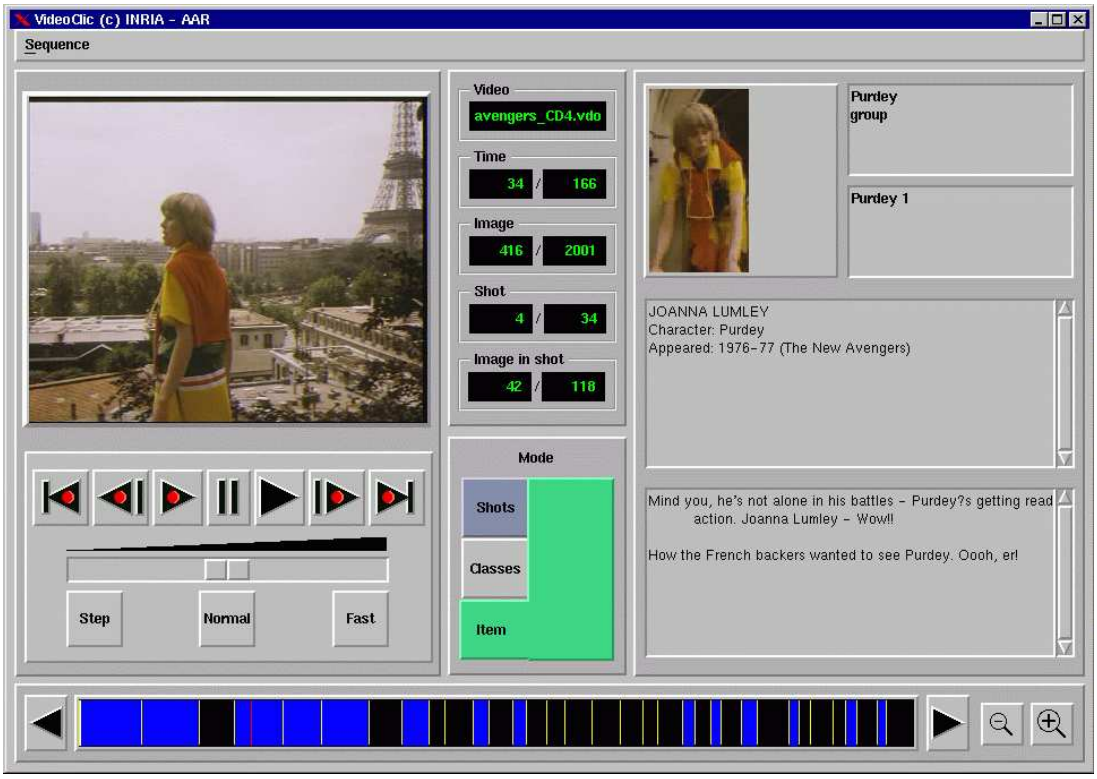


FIG. 1.2: *Application d'une vidéo interactive : l'utilisateur demande plus d'informations sur l'actrice "Purdey" dans le film "Avengers"; par un simple clic sur une de ces apparitions dans la vidéo (coin haut-gauche), des informations diverses (histoire, rôle, ...) s'affichent à coté de la vidéo et des outils de navigation avancés sont activés qui lui permettent par exemple de se positionner directement sur la scène ou l'apparence suivante/précédente de Purdey.*

sont apparus, le SMIL (*Synchronised Multimedia Interface Language*) standard des logiciels Realplayer et Quicktime, et le format ASF (*Advanced Streaming Format*), utilisé par le logiciel Windows Media Player.

L'interaction à laquelle ce travail se limite est liée à la navigation utilisant la structure de vidéo et les hyperliens mentionnés ci-dessus. Cette interaction permet non seulement une navigation en suivant les hyperliens, mais offre des extensions naturelles comme des recherches à granularité variable: parcours accéléré de scènes en scènes à la recherche d'une action particulière où apparaît un acteur donné; recherche plus détaillée d'un plan particulier à l'intérieur d'une scène.

Quelques éléments de cette spécification de l'interactivité sont offerts par des produits commerciaux en vente aujourd'hui, comme par exemple "Mvshots" et "Movideo 2 Studio" conçus par la société française Artsvidéo Interactive. Le produit Mvshots possède une fenêtre de navigation dans la structure plan-séquences, et Movideo offre en plus une interaction au niveau des objets qui ont manuellement été segmentés et rattachés à des

liens internes et externes.

1.2 Les problèmes soulevés

L'interaction entre l'homme et la vidéo se fait nécessairement au travers de la machine. Il en découle que les problèmes évoqués plus haut interviennent à deux niveaux : au niveau Homme/Machine et au niveau Machine/Vidéo. Au niveau Machine/Vidéo, l'incompréhension est totale : la machine n'est pas capable de *comprendre la vidéo* et donc a priori de la manipuler. Un traitement, globalement désigné sous le terme d'**indexation**, doit être mis en oeuvre pour structurer un document vidéo brut et en faciliter la manipulation. Au niveau Homme/Machine, c'est la représentation qui pose le plus de problèmes. L'homme est capable de *comprendre la vidéo* mais il n'arrive pas à communiquer avec une machine qui a du mal à le comprendre et à lui montrer facilement ce qui l'intéresse. Les requêtes portent généralement sur un concept sémantique entièrement absent chez la machine. Par exemple, la machine n'est pas capable d'interpréter une requête simple du genre : "Trouvez moi la scène où Mr. Bean est sur la plage".

Au niveau conceptuel de cet handicap entre l'homme et la machine se rajoute une difficulté technique dûe à la nature à la fois temporelle et spatiale de la vidéo. D'abord, la vidéo représente un volume énorme d'information (1 heure de vidéo en format compressé correspond à 575 Mega octets environ). Ensuite, les entités présentes dans cette vidéo n'ont pas de caractérisation fiable et ne sont pas extractibles : un personnage se confond plus ou moins dans la scène, et ses aspects varient considérablement durant le film. On se trouve donc dans une situation très différente du texte où un mot se localise facilement; il a un ensemble de sens très limité et très peu de variations orthographiques.

Face à cette double difficulté, nous listons ci-dessous les problèmes à résoudre pour atteindre nos objectifs :

- **Extraction des objets :** L'intérêt qu'un spectateur donne au contenu visuel d'une vidéo se focalise essentiellement sur les objets comme personnages, voitures, objets commerciaux, etc. L'extraction de ces objets, qu'elle soit automatique ou semi-automatique, représente une tâche très complexe. Ceci est dû principalement au fait que les modèles de ces objets ne sont pas connus a priori, au contraire de l'extraction des visages ou des doigts par exemple.

- **Indexation d'objets vidéo :** L'indexation de toutes les images ou objets d'une vidéo n'est pas faisable à cause du nombre énorme des images à indexer (à un film de 1h30 correspondent 129600 images). Les dimensions des descripteurs à extraire de ces images sont généralement assez grandes et celles-ci rendent inefficaces les structures de données souvent utilisées pour gérer ces descripteurs et rechercher des requêtes.

- **Appariement des objets :** Il s'agit ici d'identifier les différentes apparences du même objet. Ces apparences intra- et inter-plans vidéo sont très variables et rendent leur identification très délicate. Le changement d'apparence est dû à la nature de

la vidéo : les images sont acquises dans des conditions naturelles et le montage des scènes exige de filmer les objets (acteurs, voitures, ...) sous une grande variété de prises de vues, de changements d'échelles et d'éclairages, etc.

- **Classification des objets vidéo :** La classification est une technique qui regroupe les apparences d'objets en différentes classes d'équivalence. Une telle technique est pratiquement plus difficile que l'appariement. Car, d'une part la classification hérite les problèmes de l'appariement qui lui représente un processus "interne", et d'autre part en classification automatique il s'agit de déterminer le nombre de groupes d'objets à fabriquer, ce qui n'est pas du tout facile sur des données simulées, voir des objets d'apparences variables.

- **Segmentation élémentaire de la vidéo :** Il s'agit ici de regrouper les images successives en des unités qui correspondent à des prises de vues de la caméra souvent connues sous le nom de "plans vidéo". Notons que toutes les méthodes de traitement du contenu de la vidéo (suivis d'objets dans les plans, similitudes entre plans, etc.) se basent sur ces unités et donc la fiabilité de leurs résultats dépend de la précision de la détection des plans.

- **Sélection des images- apparences-clés :** Pour permettre une visualisation rapide du contenu d'un plan vidéo et afin de simplifier la présentation d'un très grand nombre de plans, la sélection des images ou apparences les plus représentatives du plan ou objet suivi semble être un point important à résoudre. Souvent le plan vidéo est résumé par l'image médiane. Ce choix n'est pas toujours valide surtout quand le plan n'est pas fixe et son contenu varie (objets mobiles).

- **Macro-segmentation de la vidéo en scènes :** Lorsqu'il s'agit d'un film vidéo d'une heure contenant environ par exemple 1000 plans, il devient très difficile de les représenter graphiquement et à les explorer linéairement par l'utilisateur. En effet, un découpage plus macroscopique que les plans vidéo en des unités appelées *scènes* est réalisé. Cette technique de macro-segmentation peut-être vue comme un processus qui consiste à créer une partition de l'ensemble de tous les plans d'un film vidéo. Cela veut dire que deux problèmes se présentent face à un processus de macro-segmentation : le problème de la classification automatique des plans et le problème de la définition de la sémantique lors de la classification des plans.

- **Interaction entre utilisateur et vidéo hyperliée :** Lors de la création d'une vidéo hyperliée l'intervention de l'"'auteur de la vidéo hyperliée" peut-être indispensable, par exemple aux endroits où les méthodes automatiques d'indexations échouent. A ce niveau, il s'agit de définir précisément ces interactions et de fournir des outils efficaces de présentations et d'éditions des résultats d'indexation. Aussi, lors de la présentation d'une vidéo hyperliée à un utilisateur final, il s'agit ici de fournir des interfaces graphiques simples et intuitives, qui reflètent la puissance et la souplesse d'utilisation d'une vidéo hyperliée.

1.3 Les contributions de cette thèse

Face aux problèmes cités au-dessus les contributions principales de cette thèse peuvent être résumées par les points suivants :

- **Modélisation de la variabilité intra-plan :** Nous proposons de modéliser la variabilité intra-plan d'un objet suivi d'un plan vidéo, dans l'espace de descripteurs, par une fonction statistique de *densité de mélange gaussien*, qui **capture** les différentes apparences intra-plan de l'objet. L'idée ici est de classifier les apparences intra-plan semblables en des groupes. Ensuite, chaque groupe d'apparences sera représenté par ses *paramètres statistiques*, ce qui **réduit** considérablement le nombre des descripteurs à indexer par objet suivi (par plan vidéo).

- **Identification d'objets à travers la vidéo par une classification supervisée :** Dans un premier temps, l'auteur de la vidéo hyperliée choisi d'une façon interactive les "modèles d'objets suivis". Dans un second temps, une modélisation de la variabilité de ces objets est effectuée. En adoptant le mélange gaussien dans ce cas de discrimination, deux techniques sont employées pour classer les requêtes (toutes les apparences d'objets dans le film) : le **classement individuel** par maximum a posteriori (MAP) et le **classement robuste** par vote majoritaire.

- **Identification d'objets à travers la vidéo par une classification automatique :** Pour automatiser la classification des objets suivis, on calcule des distances adaptées à nos données statistiques représentant les objets suivis : *les densités de mélanges gaussiens*. La distance entre deux objets suivis est calculée par le minimum de la distance de Kullback (ou de Bhattacharyya) entre les composantes gaussiennes des deux densités de mélanges correspondant. Sur la matrice de distances ainsi obtenue, une classification ascendante hiérarchique est effectuée. Une méthode interactive a été proposée également pour choisir le nombre de classes d'objets suivis. De même des outils graphiques et interactifs sont mis à la disposition de l'auteur de la vidéo interactive pour corriger éventuellement de mauvaises classifications d'objets ainsi obtenues.

- **Sélection des images représentatives d'un plan :** Face au problème de sélection des images- apparences-clés, et en s'inspirant des travaux précédents sur la modélisation de la variabilité intra-plan des objets suivis, il était facile pour nous de proposer une technique performante à ce problème. Après l'identification des classes d'apparences intra-plan, chaque classe d'apparences est d'abord représentée par l'apparence médiane, et ensuite deux tests de vérification temporelle et spatiale permettent de ne garder que les images- apparences-clés les plus représentatives. Ce choix des images-clés est automatique.

- **Macro-segmentation d'une vidéo en scènes :** Pour la macro-segmentation d'un film vidéo, nous avons mené d'abord une étude expérimentale sur la comparaison de deux plans vidéo sur la base de trois descripteurs de bas niveaux : le contenu du plan, la distribution globale de la couleurs et la distribution spatiale de la couleurs

de l'image médiane du plan. La distance proposée pour mesurer la similarité entre deux plans sur la base du descripteur du contenu est le nombre des objets similaires de deux plans. Cette étude a démontré l'importance d'une telle stratégie de caractérisation mixte du plan vidéo. De là, nous avons étendu une approche existante de macro-segmentation [53] [25]. Cette extension est résumée par une étape de fusion des classes de plans obtenues par les différents descripteurs. Une amélioration nette des résultats de l'approche de base est réalisée sur plusieurs séquences vidéo.

- **Implémentation de ces outils dans un prototype :** En analysant les performances et les limites des méthodes utilisées lors de la construction de la vidéo hyperliée, nous avons défini les interactions entre l'indexeur, la vidéo et la machine. Ensuite on les a implémentées dans un système de fabrication de la vidéo hyperliée appelé **VideoPrep**. Les résultats obtenus après cette étape sont utilisés par un autre système de présentation de la vidéo hyperliée nommé **VideoClic**.

  Des démonstrations pour l'ensemble des approches proposées dans cette thèse sont disponibles sur le web à l'adresse suivante :

  `http://www.inrialpes.fr/movi/pub/Demos/DemoRiad`.

1.4 Organisation de ce mémoire

Chapitre 2. Il rappelle la structure hiérarchique d'un film vidéo, ainsi que les méthodes de segmentation en plans et de suivis d'objets dont nous nous sommes servis durant ce travail.

Chapitre 3. Les problèmes d'indexation et de variabilité abordés plus haut, ainsi que l'approche retenue pour palier à ces deux problèmes, sont discutés en détail dans ce chapitre. Les données expérimentales (descripteurs de couleurs) et leur pré-traitement sont décrites ici.

Chapitre 4. Il décrit l'approche semi-automatique de classification des objets basée sur les modèles de variabilité estimés dans le chapitre précédent. Une validation expérimentale de cette approche sur la séquence vidéo "**Avengers-1**", ainsi qu'une comparaison de performance avec une méthode classique d'appariements sont également présentées.

Chapitre 5. L'objet de ce chapitre est de décrire en détail l'approche de classification automatique des objets suivis dans l'espace de paramètres gaussiens. Les expérimentations sont menées sur la séquence vidéo "**Avengers-2**".

Chapitre 6. Il détaille l'algorithme de sélection des apparences-clés d'un objet suivi.

Chapitre 7. Ce chapitre rappelle les deux étapes de base pour la macro-segmentation en scènes, ensuite il présente l'extension proposée durant cette thèse. Des résultats sur la séquence "**Dances with Wolves**" sont montrés pour l'approche de base et son extension.

Chapitre 8. Les niveaux d'interactions entre Utilisateur/Vidéo, Utilisateur/Machine et Machine/Vidéo ainsi que les deux systèmes **VideoPrep** et **VideoClic** sont l'objet de ce chapitre.

Chapitre 9. Cet ultime chapitre présente le bilan du travail effectué durant ces trois ans de thèse. Quelques perspectives de recherches et d'applications terminent ce mémoire.

Chapitre 2

Généralités sur la structuration de la vidéo

Je reglerai au plus vite
les petits pépins.

L'objectif de ce chapitre est de rappeler la structure hiérarchique d'un document vidéo et de décrire brièvement les méthodes utilisées dans ce travail pour segmenter la vidéo en plans élémentaires et pour suivre automatiquement ou semi-automatiquement des régions mobiles à travers le temps. Nous nous servirons des méthodes développées par l'équipe Vista de l'Inria Rennes suite à une collaboration entre trois partenaires : Alcatel Alsthom Recherche et les deux projets de l'Inria Movi et Vista. Après une courte présentation de ces outils nous évaluons leurs performances sur des exemples réels. Ensuite, nous décrirons le corpus vidéo utilisé dans l'évaluation de nos approches, et avant de conclure on introduit le future standard MPEG-7.

2.1 Structure hiérarchique de la vidéo

Plusieurs catégories des documents vidéo existent dans la réalité : les journaux télévisés, les émissions sportives (foot, tennis, etc.), les dessins animés, les films, etc. Parmi eux il y en a qui possèdent un modèle a priori (e.g. les journaux télévisés) et d'autres qui possèdent une histoire (e.g. les films). L'analyse des émissions de journaux télévisés définit le document comme étant une alternance entre une séquence d'images montrant le présentateur et des séquences d'images associées à des reportages [132]. Les plans sont classifiés selon un modèle spécifique à chaque type de plan : début, présentateurs, publicités, météo et fin.

Si on considère la structuration et la présentation des films cinématographiques, ceux-ci ne disposent d'aucune information a priori sur leur présentation. Leur traitement devra donc être très général et pourra donc servir à tout autre type de documents vidéo.

Un film n'est pas simplement une suite quelconque d'images, il récite une histoire et donc il est possible d'y associer une structure de composition hiérarchique [85] [68]. En général, cette histoire est décomposée en plusieurs épisodes ou *séquences narratives* et chaque épisode est formé de plusieurs *scènes* ou actions. A son tour chaque scène regroupe une suite continue dans le temps de segments d'images ayant un rapport sémantique entre eux. Ces segments sont souvent connus sous le nom des *plans cinématographiques*.

Les différents niveaux physiques et sémantiques qui peuvent être distingués dans la structure d'un film sont :

- **Plan vidéo :** une suite d'images filmée sans interruption temporelle par la même caméra. Il représente l'unité physique de base pour la construction et la manipulation de la vidéo.

- **Image clé :** une image du plan vidéo qui peut représenter visuellement son contenu.

- **Scène vidéo :** est définie comme étant une collection de plans, adjacents dans le temps, ayant un lien sémantique entre eux et qui possèdent une unité d'action et de lieu.

- **Groupe vidéo :** une entité intermédiaire entre les plans physiques et les scènes sémantiques. Dans la cas d'une vidéo segmentée en régions on peut distinguer entre "groupe de plans" et "groupe d'objets".

- **Groupe de scènes :** une suite de scènes non successives où l'unité de lieu est conservée.

- **Séquence narrative :** une suite de scènes successives où l'unité d'action est conservée. Deux séquences narratives sont souvent séparées par un effet de transition de type *fondu au noir*.

La figure 2.1 illustre ces différents niveaux et les liens entre eux. Les niveaux *plan*, *scène* et *séquence* sont considérés comme des niveaux de base pour la production et l'analyse des films [68] [50]. Rui et al. [105] dans leur contribution à MPEG-7 proposent une structure similaire à celle de la figure 2.1. Au niveau "groupe vidéo" ils fabriquent des groupes de plans locaux (dans un intervalle de temps) pour ensuite construire les scènes. Donc, pour eux, c'est un niveau caché à l'utilisateur final de la vidéo structurée. Par contre, dans ce travail, et comme nous le verrons dans les chapitres suivants, les groupes d'objets (plans) sont des entités essentielles pour une vidéo structurée et interactive. Aussi, à partir de deux niveaux "groupe vidéo" et "scènes" on déduit un nouveau niveau "groupe de scènes" qui n'a pas les mêmes caractéristiques que le niveau "séquence narrative".

Un document vidéo structuré comportera à la fois des informations liées à la structure du document, comme la position des plans et les relations qui les unissent pour former les scènes, et des informations relatives au contenu des plans, comme la description des objets en présence et les relations de similarités.

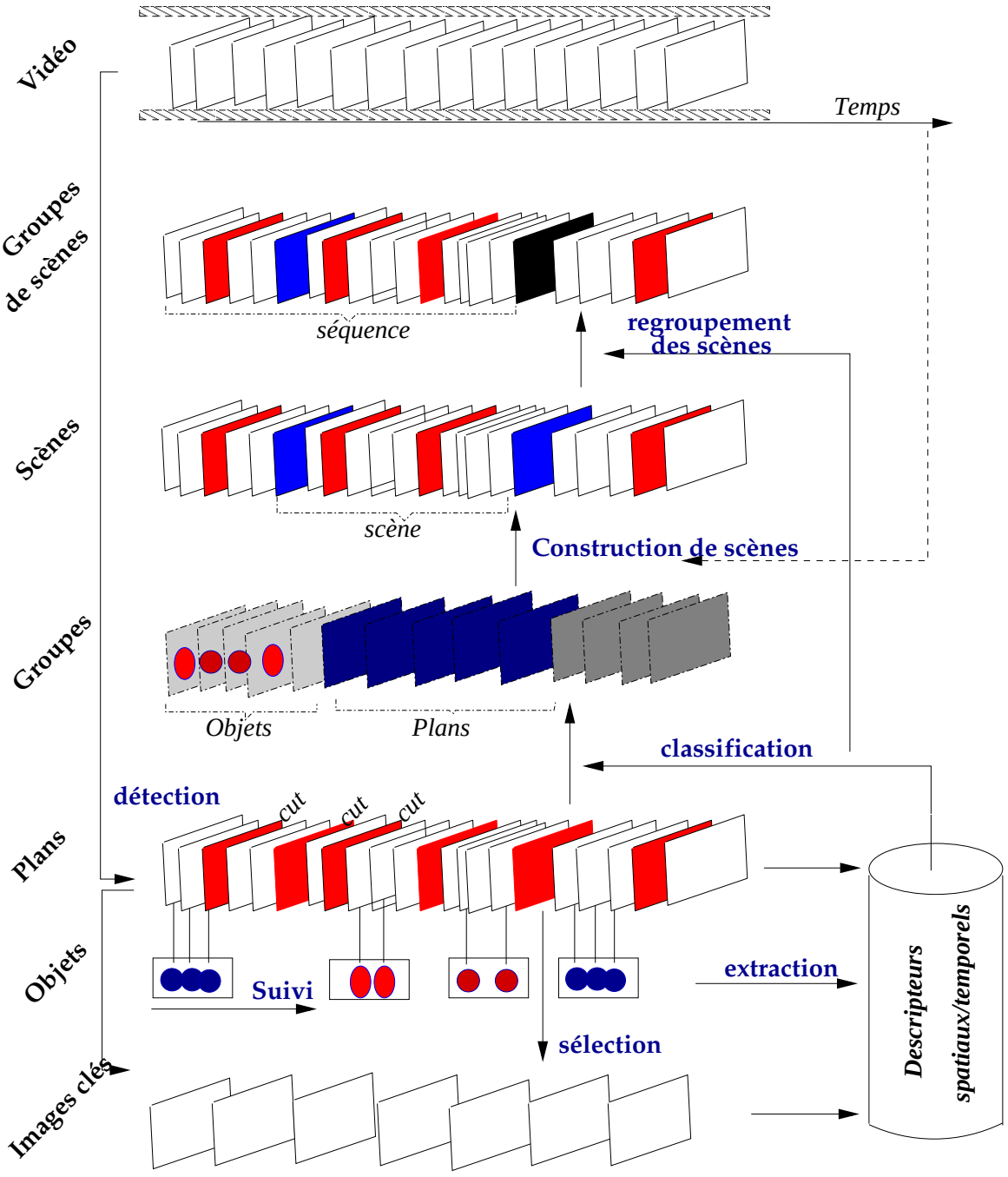


FIG. 2.1: *Une représentation hiérarchique de la structure de la vidéo*

2.2 Vers une structuration automatique

Définition 2.2.1 *Structuration : La structuration est un processus de segmentation spatio-temporelle des documents en segments de différentes unités et de construction des relations entre ces segments. En d'autre termes, c'est une opération d'abstraction dont le but est de trouver les différentes corrélations sémantiques au cours du temps.*

• **Aperçus des problèmes :** L'handicap majeur de l'utilisation de la vidéo est dû d'une part à sa taille importante et à son format purement séquentiel, et d'autre part à la difficulté d'en extraire une information sémantique à la différence du texte.

A l'heure actuelle la production et la diffusion de la vidéo numérique au niveau mondial est gigantesque. Par exemple, l'Inathèque de France[1] accumule environ 17.000 heures par an d'émissions télévisées [65] soit 85.000 heures depuis son ouverture (en Janvier 1995). Aussi, si on prend l'exemple important des bases d'images et vidéos constituées sur les serveurs web du monde entier, le volume du contenu accessible et la rapidité de son évolution sont considérables.

Du fait que la structure d'un film n'est pas fournie a priori, le seul moyen de le consulter est de se servir des outils classiques de navigation et de recherche des lecteurs vidéos : lecture arrière et avant accélérée, à vitesse normale ou ralentie, l'arrêt sur l'image et un ascenseur permet, à l'aide d'un curseur, un positionnement par accès direct aléatoire dans le document.

• **La structuration :** Pour permettre de gérer la taille importante de la vidéo et d'explorer son contenu d'une façon non séquentielle et efficace, la structure intrinsèque de la vidéo doit être identifiée. Comme nous l'avons mentionné dans l'introduction, par analogie avec les livres, la structure hiérarchique peut servir de table de matières pour la vidéo.

La structuration manuelle de la vidéo semble être inefficace : quantité immense de données, coût de la main d'oeuvre, subjectivité et nature monotone et fatigante des travaux. En conséquence, la recherche dans ce domaine est orientée depuis quelques années vers des méthodes automatiques de structuration et d'indexation par le contenu [12] [129] [53]. La figure 2.2 résume les points essentiels de la structure d'un système de vidéothèque adapté par Yeo et Yeung [129].

La suite de ce chapitre se focalise sur les méthodes dont nous nous sommes servi pour segmenter la vidéo en plans et pour localiser et suivre des objets à travers le temps.

2.3 Segmentation temporelle en plans

• **Principe :** La détection des ruptures de plans est basée sur la segmentation ou partitionnement de la bande vidéo en unités adjacentes simples. Un plan, défini par la succession de plusieurs images représentant une action spatiale et temporelle continue, est considéré comme l'unité de base de manipulation de la vidéo (figure 2.3). Le partitionnement de

1. département de l'Institut National de l'Audiovisuel (INA) chargé de la mise en oeuvre du dépôt légal de la radio-télévision française

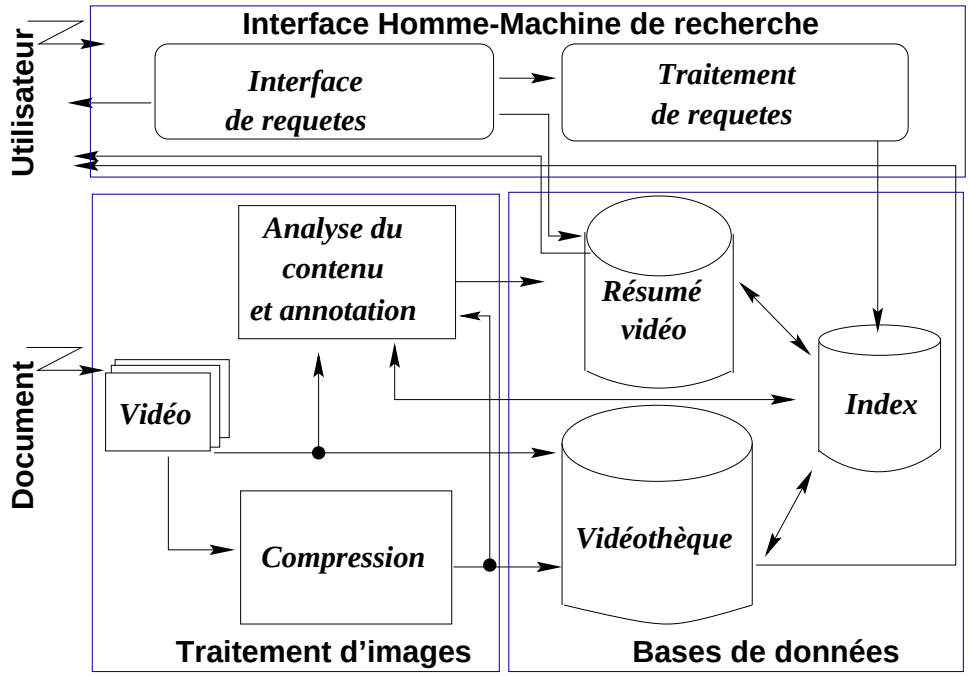


FIG. 2.2: *Schéma de la structure d'un système de vidéothèque.*

la vidéo est peut être vu comme un processus de recherche de rupture ou d'anomalie de continuité d'un certain critère quantitatif. Par exemple, la différence entre deux images consécutives constitue un critère quantitatif simple qui peut être utilisé pour détecter des ruptures brusques de plans.

• **Les méthodes existantes :** Actuellement, le problème de détection des ruptures de plan est bien maîtrisé et plusieurs méthodes efficaces ont été proposées dans la littérature [4] [122] [13] [14]. Ces méthodes peuvent être subdivisées en deux grandes catégories : celles qui cherchent à évaluer une *similarité* entre images successives et d'autres qui cherchent plutôt à évaluer si le contenu à $t+1$ est une suite *cohérente au contenu* de l'image t.

Parmi les méthodes de la première catégorie on peut distinguer entre les méthodes qui analysent la distribution des couleurs en utilisant des histogrammes globaux [89] ou locaux [72], les méthodes basées sur la comparaisons de primitives extraites des images comme le contour [131] et les foyers d'expansion [4] (section 2.3.2), et les méthodes qui analysent directement la bande compressée MPEG-1 [128] [24].

La deuxième catégorie regroupe les méthodes qui examinent la manière dont une information liée au mouvement estimé dans la séquence peut décrire "le degré de continuité" du contenu [14] (section 2.3.1).

Une comparaison quantitative des performances des méthodes existantes est difficile, chacun des travaux utilisant son propre jeu de tests. Une étude comparative récente est menée par [40]. Parmi les méthodes évaluées, les meilleures performances sont obtenues par [128], mais elles restent très moyennes, notamment en ce qui concerne la détection des

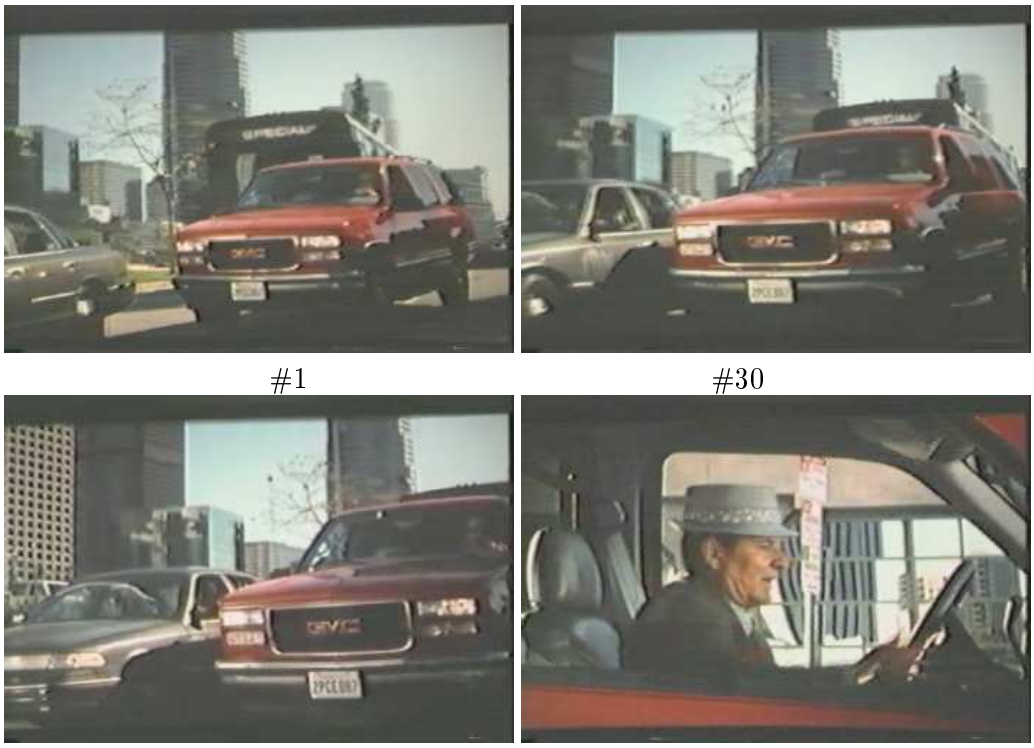


FIG. 2.3: *Exemple d'un changement de plan dans une séquence vidéo.*

transitions progressives (fondu enchaîné, fondu au noir, etc.).

• **Critères de choix d'une méthode :** Le choix d'une méthode de découpage de la vidéo en plans dépend des besoins de l'application en terme de rapidité de calcul et de précision des résultats. Les techniques assez simples mènent à des performances pouvant satisfaire un certain nombre d'applications importantes qui réclament surtout la rapidité de traitement. Le traitement de la représentation compressée MPEG est alors une option pertinente. D'autres applications peuvent être moins exigeantes en temps de calcul, mais nécessitent des résultats complets et précis. Par exemple, la qualité de la fabrication des scènes dépend de la pertinence des résultats de segmentation en plans et de classification de ces plans. Donc un facteur à prendre en compte aussi pour choisir une méthode de découpage en plans est la présence éventuelle d'autres étapes d'analyse dans le système, généralement en aval. En effet, si ces étapes utilisent le résultat du découpage en plans, il faut étudier son impact sur les modules qui en dépendent. D'autre part, il faut évaluer le coût calculatoire du découpage en plans, relativement au coût total de calcul d'un système d'analyse du contenu - et le coût de correction manuelle.

Dans la section suivante nous présentons brièvement la méthode de Bouthemy et al. [14] employée dans ce travail suite à une collaboration déjà mentionnée au début de ce

chapitre. Les avantages et les inconvénients de cette méthode sont aussi rappelés. Ensuite, nous présentons la méthode de Ardebilian et al. [4] dont nous nous somme servis dans la segmentation en plans de la séquence "Danse avec les loups" pour un but de construction des scènes [52] [51] (chapitre 7).

2.3.1 Mesure de cohérence temporelle du contenu

Cette méthode proposée par Bouthemy et al. [14] est basée sur l'estimation robuste d'un modèle affine de mouvement dominant. Le modèle affine (équation 2.1) permet de se rendre compte d'une large variété de mouvements $2D$ (translation, rotation, divergence, etc.), dont l'estimation peut se faire d'une manière fiable.

$$\overrightarrow{w}_{\ominus}(p_i) = \begin{pmatrix} a_1 + a_2(x_i - x_g) + a_3(y_i - y_g) \\ a_4 + a_5(x_i - x_g) + a_6(y_i - y_g) \end{pmatrix} \tag{2.1}$$

où $\overrightarrow{w}_{\ominus}(p_i)$ représente le vecteur de déplacement à un point $p_i = (x_i, y_i)$ du point référence (centre de gravité du support). Les paramètres du modèle $\ominus = (a_1, ..., a_6)$ sont estimés par la technique de moindres-carrés pondérés itérés.

Après l'estimation d'un tel modèle pour une image donnée l'ensemble total des pixels dénoté par S est divisé en deux sous ensembles : pixels "conformes" (S_c) ou "non-conformes" au modèle de mouvement. Une comparaison des valeurs de w_i avec un certain seuil permet de juger si un pixel est conforme ou non au modèle. A partir de ça ils définissent une mesure de "cohérence inter-images"

$$\Im_t = \frac{n_c(t)}{n(t)} \tag{2.2}$$

où n et n_c représentent les cardinaux respectifs des ensembles S et S_c.

Cette mesure est ensuite exploitée pour détecter les changements de plans. Pour deux images successives appartenant au même plan, le support d'estimation est de taille importante. Par contre, dans le cas d'une paire d'images aux contenus très différents, le modèle de mouvement estimé ne sera pas capable d'expliquer la transformation entre les deux images, et donc la taille du support d'estimation sera faible. Le test cumulatif de Hinkley est utilisé plutôt qu'un simple seuillage sur les valeurs de $\Im_t$ afin d'améliorer la détection de changement de plan et de distinguer entre différents types de transition entre deux plans (cut, fondu, volet, etc.).

L'avantage de cette technique est qu'elle permet de différencier un changement de contenu dû à un mouvement de la caméra d'un changement induit par un effet de transition progressive (fondu par exemple). Par contre, le modèle affine $2D$ utilisé par cette technique peut être mis en défaut et le support du mouvement estimé est de taille faible, notamment en présence des grandes variations d'illuminations et lorsque le fond de la scène est constitué de plusieurs grands éléments subissant des mouvements différents, ou d'éléments non-rigides.

2.3.2 Méthode d'analyse de la scène 3D

En 1996, à la différence de toutes les méthodes statistiques de détections de plans citées ci-dessus, Ardebilian et al. proposent une méthode qui touche à la structure géométrique de l'image [4].

L'idée de base consiste à approximer l'image par des segments de droites. Cela suppose que l'image est segmentée préalablement en éléments pertinents susceptibles d'êtres approximés par des droites. L'image de contours est calculée premièrement à partir d'une carte de gradient. Ensuite, une approximation communément appelée *line fitting* des points de contours est réalisée à l'aide de la transformation *point-to-line* de Hough [61]. La transformée de Hough permet, en changeant d'espace de paramètres, de détecter des droites et d'en donner une équation. Elle est basée sur la dualité de l'équation d'une droite

$$x \sin\theta - y \cos\theta = \rho \tag{2.3}$$

En effet, une droite peut être définie soit par un couple (ρ, θ) soit par deux points de l'image, l'équation 2.3 permettant le passage d'une représentation à l'autre.

Les indices extraits d'une image sont représentés par les points d'intersections des droites obtenues après une deuxième application de la *transformée de Hough* sur la trace du contour de l'image. Le but de la deuxième transformée de Hough est de réduire au minimum le nombre des droites qui approximent l'image. Une détection de plan est ainsi signalée lorsqu'un changement significatif du nombre de ces indices 3D entre deux images successives est réalisé. Les transitions progressives ne sont cependant pas considérées par cette étude.

Un taux de plus de 95% d'identifications correctes des "cuts" est réclamé par les auteurs de cette méthode. Cependant, il parait bien que la méthode est sensible aux occultations et/ou apparitions des objets (mobiles) dans la scène qui conduisent à des variations majeures dans l'image de contours. Sur l'exemple de la figure 2.3 il est très probable d'avoir détecté un changement de plan avant l'image 90.

2.4 Suivi d'objets dans la vidéo

Le suivi d'objets quelconques dans une séquence vidéo réelle est une tâche très délicate, surtout quand ces objets sont non-rigides, le fond de la scène n'est pas fixe et dans le cas de plusieurs objets mobiles dans la même scène. Lorsqu'il s'agit de suivre des objets particuliers comme le visage, la main, le bras, etc. la tâche est plus simple car le modèle et les contraintes de variations de ces objets sont connus a priori [95] [82] [114]. D'autres critères de type géométrique et/ou statistique peuvent aussi être introduits dans l'identification des ces objets [16] [64].

Le suivi d'une primitive d'un instant t vers un instant $t+1$ est généralement basé sur une prédiction de la position de cette primitive recherchée à l'instant $t+1$. Dans ce mémoire on utilise le terme apparence pour désigner une occurrence de l'objet suivi à un instant t du plan. Dans le système conçu pour la construction de la vidéo interactive [8] [56] (chapitre 8), deux techniques de suivis d'objets *automatique* et *semi-automatique* sont intégrées. Ces techniques ont été proposées par l'équipe Vista [42].

2.4.1 Approche automatique

Nous rappelons ici les grandes lignes de cette approche. Pour plus de détails le lecteur peut s'adresser a la thèse de Gelgon [41] (première partie) ou bien à [42].

Cette approche peut être résumée en deux étapes principales :

1. La première étape consiste à chercher une partition de l'image au sens du mouvement. Dans un premier temps une partition spatiale de l'image est calculée. L'estimation de cette partition est effectuée à l'aide d'une méthode classique de segmentation d'images : ils supposent que la distribution de niveaux de gris des pixels de l'image est un mélange de lois gaussiennes pour lequel l'algorithme EM pourra être appliqué. Ensuite on applique une procédure de fusion des régions adjacentes dans l'image basée sur un critère de minimisation d'énergie entre ces régions. La fonction d'énergie est exprimée en fonction des descripteurs extraits des régions. Dans un deuxième temps, les régions spatiales (de la nouvelle partition ainsi obtenue) sont caractérisées par des modèles de mouvements 2D (affine à 6 paramètres). Cette caractérisation permet de définir une mesure de cohérence de mouvements entre deux régions spatiales. Cette mesure, prenant en compte la *continuité* du champ de mouvement à la frontière des deux régions, évalue la différence entre les deux champs de vecteurs de vitesse étendus au support formé par l'union des deux régions. Une étiquette de mouvement est associée à chaque région.

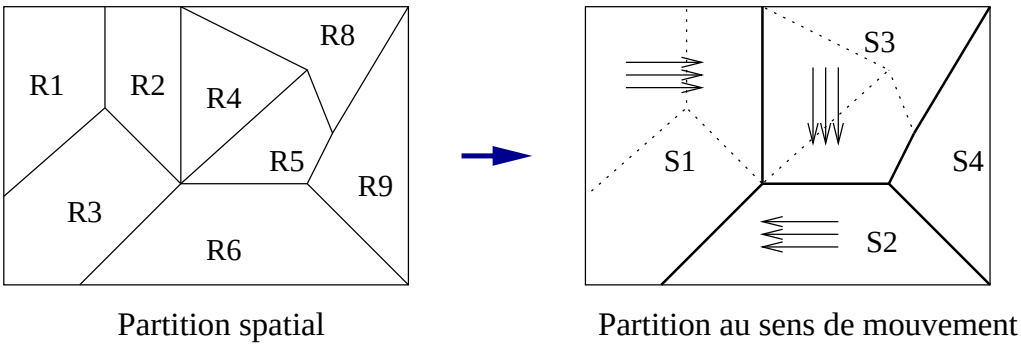


FIG. 2.4: *Partition au sens de mouvement*

2. La deuxième étape est basée sur un principe de propagation des étiquettes de deux phases : prédiction et mise à jour des configurations d'étiquettes définissant deux partitions à l'instant t et $t+1$. La prédiction de la configuration d'étiquettes construite à l'instant t est utilisée comme configuration initiale d'étiquettes, relative à la segmentation spatiale, à l'instant $t+1$. La mise à jour d'une configuration d'étiquettes revient à supprimer ou rajouter des régions à l'instant $t+1$.

L'utilisation du mouvement estimé par l'union des régions spatiales suppose qu'un modèle affine soit à même de la décrire, ce qui n'est pas nécessairement vrai. Ceci peut mettre en défaut une telle approche. D'un autre coté la méthode est basée sur une segmentation de l'image en régions dont le nombre de régions et la qualité de la segmentation

sont difficiles à être parfaitement estimés. Une illustration des résultats de cette approche est présentée dans la figure 2.5.

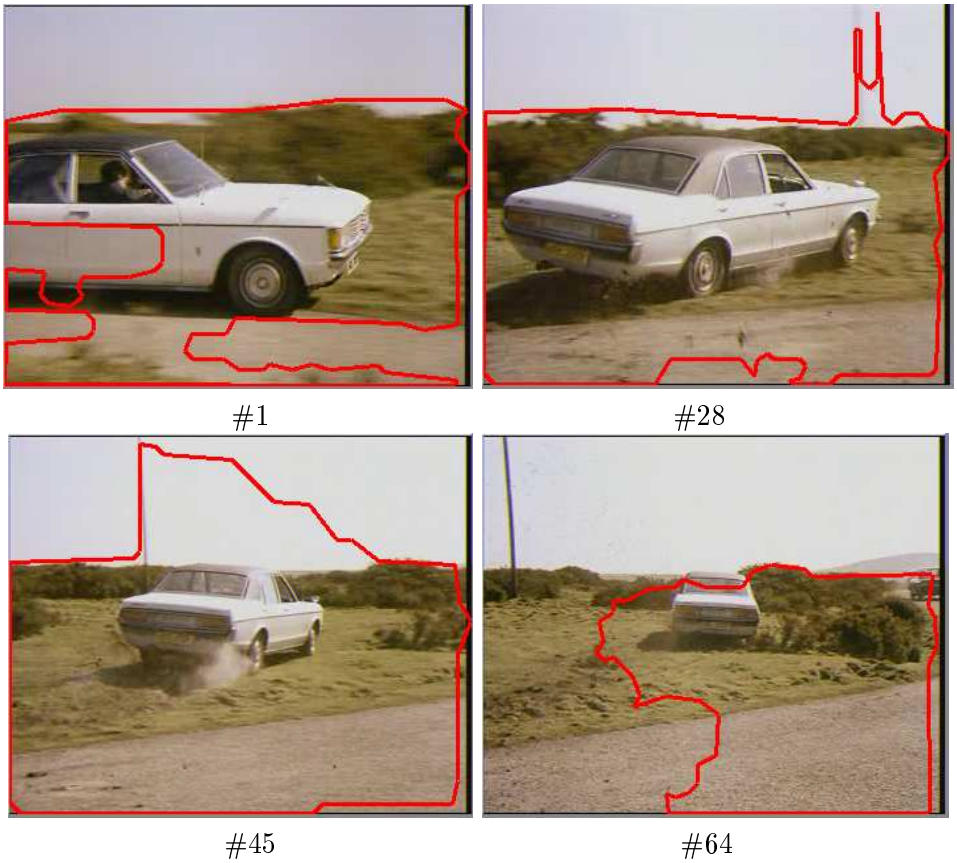


FIG. 2.5: *Résultat de suivi par l'approche automatique de [42] sur la séquence "Ford blanche" de* 66 *images.*

2.4.2 Approche semi-automatique

Lorsque le contenu de la scène est trop complexe pour l'approche automatique et quand un objet d'intérêt pour l'utilisateur n'est pas détecté automatiquement, une approche semi-automatique de suivi d'objets est avantageuse. L'approche possède une étape d'initialisation: la zone d'intérêt est définie manuellement dans une image du plan vidéo (suivi arrière, avant, et/ou dans les deux sens). Ensuite, pour chacun des couples d'images successives, la procédure suivante est alors effectuée. Entre les instants t et $t+1$, un modèle affine $2D$ du mouvement dominant sur cette zone est estimé (voir section 2.3.1). Les sommets de la zone sont projetés dans le sens de mouvement, et le polygone projeté définit la zone à $t+1$. Le nombre de sommets du polygone reste constant dans le temps. En cas de "décrochage du suivi" une alarme est fournie à l'utilisateur et de nouveau la zone d'intérêt est définie.

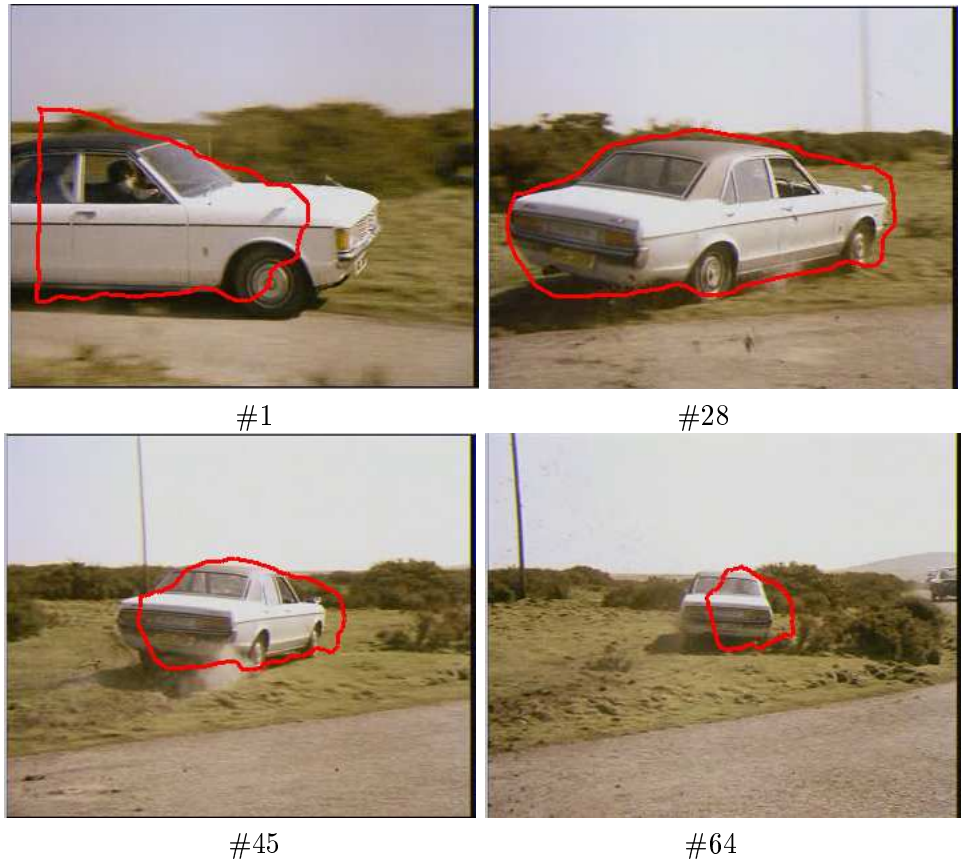


FIG. 2.6: *Résultat de suivi par l'approche semi-automatique de [41] sur la séquence d'images "Ford blanche". Le suivi de la voiture est initialisié manuellement à l'image* 28.

2.5 Corpus d'expérimentation

Les expérimentations que nous avons mené durant cette thèse pour valider les approches proposées dans les chapitres suivants, ont eu lieu sur différentes séquences du film "Chapeau Melon et Bottes de Cuir", *Avengers-1* et *Avengers-2* de la série télévision connue sous le nom *Avengers*. Ce film est fourni par l'Institut Nationale de l'Audiovisuelle (INA) comme un support standard d'évaluation des méthodes d'indexations de la vidéo en France.

Les séquences *Avengers-1* et *Avengers-2* comportent 1391 et 2749 apparences d'objets respectivement. Elles seront détaillées dans les sections 4.4 et 5.4. Ces séquences de test ont des durées assez limitées. Cette limitation est le fait de deux handicaps majeurs :

– Les outils de suivi d'objets que nous venons de présenter ne fournissent pas une segmentation parfaite des objets (voir figures 2.5 et 2.6), donc une correction manuelle du suivi et/ou un suivi manuel des objets dans les plans sont les moyens pour résoudre le problème fondamental étudié dans le contexte de cette thèse : "la

variabilité de l'apparence intra-plan des objets suivis". Le coût de cette fastidieuse correction a donc limité notre base expérimentale. Cette correction manuelle reste cependant imprécise pour nous placer dans le cas réaliste d'un futur suivi de bonne qualité, mais qui sera toujours loin d'être parfait.

– Les données visuelles (images, objets segmentés) demandent un espace disque gigantesque (environ 1.04 Giga octets pour une 1 minute de vidéo MPEG) : Les outils de préparation de la vidéo utilisés dans ce travail (segmentation en plans, suivi des objets, caractérisation de bas niveaux des images/objets) exploitent des séquences MPEG décompressés aux formats PPM [2] de haute qualité; les résultats de suivis d'objets sont aussi stockés aux formats PPM; à chaque image segmentée une image de masque lui correspond. Aussi, quelque descripteurs utilisés dans notre système comme les invariants différentiels proposés par Schmid et Mohr [110], sont extraits des images de format PGM (*portable gray map*) de niveaux de gris. Donc, pour une séquence vidéo MPEG de durée d'une minute (équivalent à 1500 images PPM), il faut un espace disque d'environ 1 Giga octets [3], si on considère qu'un objet au moins est localisé dans chacune des images décompressées et sans compter la taille des imagettes des objets segmentés (format PPM et GIF), ni la taille des descripteurs de bas niveaux extraits des objets segmentés. La conception d'une architecture travaillant sur des données compressées est donc à l'ordre du jour, mais hors du cadre de ce travail.

Par contre, ce corpus expérimental est suffisant pour valider nos approches dans des conditions réelles de changements d'images et d'apparences d'objets très variables.

2.6 MPEG-7 : interface pour la description du contenu multimédia

La future norme MPEG-7, annoncée pour septembre 2001, porte essentiellement sur la standardisation des descripteurs associés au contenu des documents audiovisuels [87]. Cette norme associe à un document audiovisuel des descripteurs, à la fois de bas niveau (décrivant les formes, couleurs, ...) et des descripteurs sémantiques pour lesquels l'intervention humaine est indispensable. Par contre, MPEG-7 ne s'intéresse ni à la définition des algorithmes d'extraction des descripteurs, ni à la manière avec laquelle on doit construire un moteur de recherche capable d'exploiter les descripteurs. La figure 2.7 illustre le domaine d'intérêt de MPEG-7.

Plusieurs groupes travaillent aujourd'hui autour de la standardisation de MPEG-7; leurs efforts se focalisent essentiellement sur les trois points suivants :

1. Normalisation d'un ensemble de descripteurs (D) où chaque descripteur sert à représenter un aspect particulier du document audiovisuel (effets sonores, couleur, texture, mouvement, suivis d'objets, etc.).

2. Portable pixel map, image 352x288, de taille 304143 octets.
3. $[(1500 \times 304143) \times 2 + (1500 \times 101391)]/(2 \times 1024) = 1039565.918$ octets.

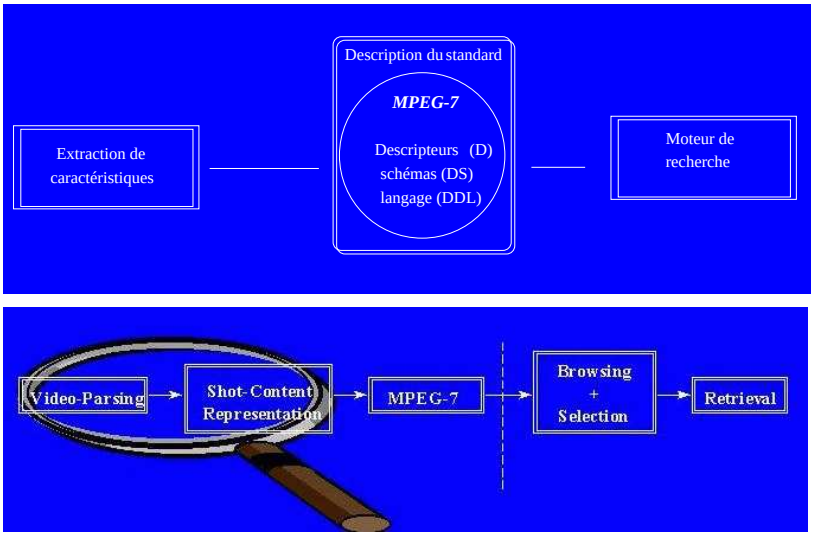


FIG. 2.7: *Le champ d'action du groupe MPEG-7.*

2. Collection de schémas de description (DS) où chaque schéma de description spécifie la structure et la sémantique des rapports entres les éléments qui composent le schéma.

3. Définition d'un langage de description (DDL) qui permet de mettre à jour les schémas et l'ensemble de descripteurs.

Une description MPEG-7 est formée par un schéma (DS) ainsi que par des instances de chaque descripteur et de chaque schéma définis dans le schéma en question (il est possible d'inclure des schémas dans un autre schéma).

MPEG-7 n'est pas encore un standard et donc l'information autour de lui n'est pas stable. Mais cet effort de normalisation semble essentiel, dans la mesure où les informations sont géographiquement très distribuées, et que de nombreux prototypes industriels d'indexation concurrents sont en cours de développement.

2.7 Récapitulatif du chapitre

Dans un premier temps, nous avons rappelé la structure hiérarchique d'un film vidéo, tout en proposant une extension de cette hiérarchie en définissant deux niveaux supplémentaires : *groupes d'objets* à travers la vidéo et *groupes de scènes* non successives dans le temps. Une telle extension pourra être suggérée pour le prochain standard MPEG-7. La construction du niveau *groupes d'objets* est discutée dans les chapitres 4 et 5. Les niveaux *images clés* et *scènes* feront l'objet des chapitres 6 et 7 respectivement.

Dans un second temps, nous avons présenté les outils de base utilisés durant cette thèse pour segmenter la vidéo en plans et pour suivre automatiquement ou semi-automatiquement les objets dans les plans. Les limites de ces outils sont présentées ainsi que leurs influences sur la démarche expérimentale de nos approches proposées.

Chapitre 3

Modélisation statistique de l'apparence intra-plan des objets vidéo

Il faut savoir
perdre du temps pour en gagner.

Dans le chapitre précédent, nous avons présenté les outils de base que nous utilisons pour partitionner la vidéo en plans, pour localiser – automatiquement ou manuellement – des objets dans les images, et pour les suivre à travers le temps.

Ce chapitre présente principalement une étude du problème de la variation de l'apparence des objets suivis à travers le temps, et les conséquences de cette variation sur la reconnaissance des objets vidéo. Ensuite, il expose l'approche que nous avons retenue pour la modélisation statistique de cette variation dans l'espace de descripteurs. Les modèles statistiques issus de cette modélisation seront utilisés dans la suite de ce travail pour indexer, apparier, et classifier les objets suivis d'une séquence vidéo.

3.1 Introduction

Le contenu dynamique d'un film vidéo au sens d'animation et mobilité des objets à travers le temps (acteurs, voitures, etc.) représente le point clé qui attire l'attention des spectateurs d'une part et les analystes de la vidéo d'autre part. L'illusion d'animation des scènes est due à la projection à grande vitesse des images fixes (de 24 à 32 images par seconde). Pour rapprocher les actions du film à la réalité, les scènes sont réalisées dans des conditions naturelles d'éclairage et d'environnement. Aussi, les acteurs d'une scène sont filmés sous une grande variété de prise de vues, de zooms et de formes.

3.1.1 Les problèmes adressés

• **Apparence et variabilité intra-plan des objets suivis** Partons d'un film vidéo partitionnée au préalable en plans cinématographiques et en objets. Les objets sont suivis dans les plans. Du point de vue vision artificielle par ordinateur, les changements d'apparence des objets les plus fréquents dans un film vidéo peuvent être résumés comme :

- *Changement de point de vue de l'objet* suivi à travers le temps, à l'intérieur des images d'un plan vidéo et/ou dans différents plans. Un changement de point de vue de l'objet correspond à une rotation $3D$ (3 degrés de liberté) et/ou à une translation (3 degrés de liberté). Les rotations $2D$ des objets dans l'image (3 degrés de translations et 1 seul degré de rotation) sont rares dans la vidéo.

- *Changement d'échelle de l'objet* d'un plan à un autre sans connaissance a priori du degré de zoom.

- *Changement de la scène* ou *Occultation partielle* de l'objet non-rigide suivi à travers le temps. Des parties sont occultées et d'autres dévoilées durant le mouvement de l'objet suivi dans le plan ou durant son apparition dans différents plans du film.

- *Changement d'éclairage* des scènes de la vidéo (éclairage artificiel ou naturel, direct ou indirect). L'obscurité de la nuit et les ombres portées ne correspondent généralement à aucun modèle d'éclairage mathématique : surfaces et reflectances sont trop complexes et surtout inconnues.

- *Changement des conditions d'enregistrement* qui inclut le flou de l'image et bruits du signal. Généralement, ces changement ne sont ni contrôlables ni prévisibles.

En réalité les objets mobiles d'un film vidéo sont contaminés par une combinaison de ces différents types de changements d'images. La figure 3.1 illustre quelques apparences intra- et inter-plan d'un objet non-rigide et en mouvement (la voiture blanche entourée). Ces images sont extraites du corpus vidéo *Avengers* (section 2.5).

Sous ces conditions difficiles, mais réelles, de changements de l'apparence des objets, il est évident que la reconnaissance des objets similaires inter-plan est une tâche très délicate. Aussi à l'intérieur d'un plan vidéo il est souvent difficile d'identifier visuellement les occurrences d'un même objet suivi. Par exemple la première ligne de la figure 3.1 met en évidence les grandes variations de point de vue et la grande variété d'occultations qui peuvent survenir.

La représentation spatiale d'un objet suivi dans l'espace de descripteurs est directement influencée par ces changements d'apparences intra-plan. Supposons que chaque occurrence de l'objet suivi est représentée par un seul point (vecteur de descripteurs) dans un espace de descripteurs multidimensionnels. Aux changements d'apparence de l'objet d'un instant à un autre correspondent une trajectoire ou un nuage de points dispersés dans l'espace. Dans la suite cet phénomène est appelé "**variabilité de l'objet suivi dans l'espace de descripteurs**". Selon la progression du changement d'apparence de l'objet à travers le temps une certaine continuité de la trajectoire peut être conservée ou non.

FIG. 3.1: *Exemple d'apparences de la voiture suivie dans la séquence vidéo Avengers*

La figure 3.2 illustre la variabilité intra-plan d'un objet mobile dans l'espace de couleurs. Dans cette séquence de publicité "Ajax" la caméra est fixe. L'objet d'intérêt est mobile : l'enfant court et se déplace vers la caméra. Donc au fur et à mesure qu'il se rapproche d'elle une partie de son corps est occultée pour qu'à la dernière image du plan, seule sa tête soit visible. Aussi, l'éclairage du soleil est bien présent sur les images du début du plan et totalement absent sur les images de la fin. La figure 3.2.a représente les quatre vues de l'enfant aux instants $1, 10, 16$ et 25 parmi 26 vues en total dans le plan. La figure 3.2.b représente la distribution des couleurs de chacune de ces quatre vues (chaque pixel est représenté par ces trois valeurs de couleurs : rouge, verte et bleu). Sans aucune idée sur les images de l'enfant, cette dernière figure permet à un analyste de conclure que ces distributions correspondent au même objet image, mais sous différentes conditions de changements d'images. D'autre part, sur la figure 3.2.c on remarque que les couleurs dominantes sont placées dans des positions différentes. Cela explique les occultations partielles et/ou les changements d'éclairages que l'enfant a subit à travers le temps. Une couleur dominante peut être vue comme étant la couleur ayant la fréquence la plus élevée. Chaque représentation $3D$ de la figure 3.2.c correspond à un histogramme calculé dans l'espace de couleur RGB de l'enfant a l'instant t ($t = 1, 10, 16, 25$). Chaque couleur (R, G, B) est représentée par un carré dont la taille est proportionnelle à la valeur de sa fréquence. La figure 3.2.d illustre cette variabilité sur la seule dimension du premier axe principal de l'histogramme RGB par rapport au temps (instants [1..26]). Visuellement, il est clair

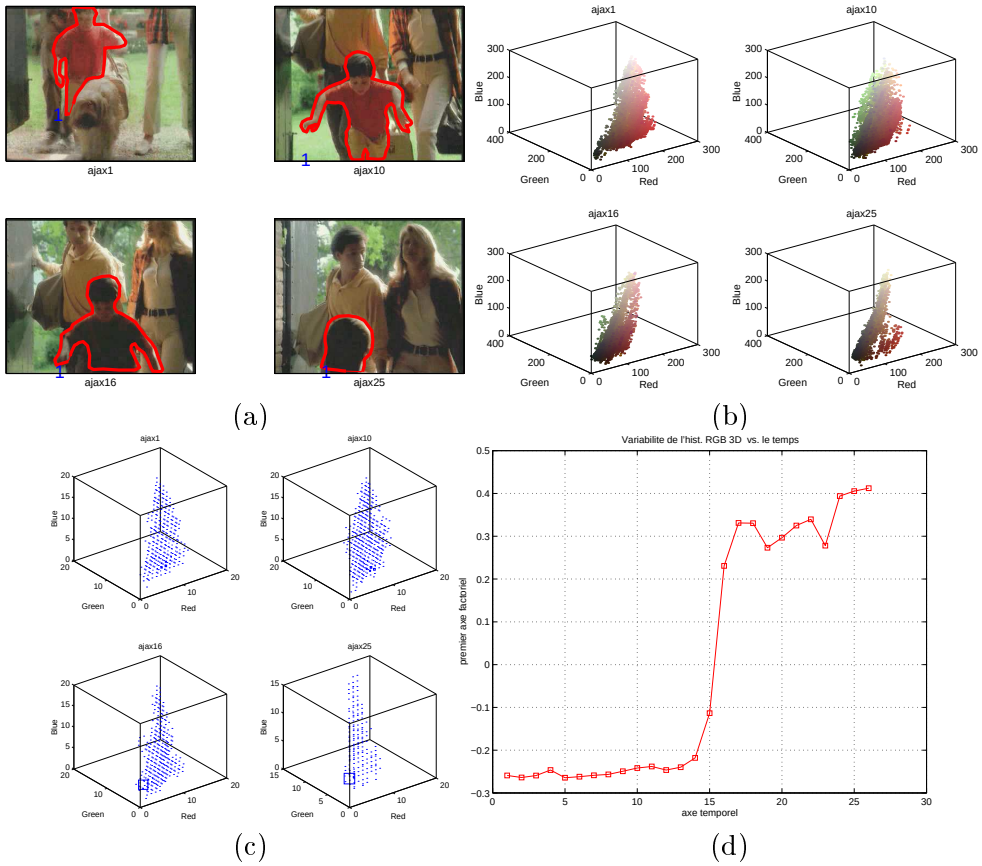


FIG. 3.2: *Exemple de la variabilité intra-plan de l'objet suivi dans l'espace de descripteurs : (***a***) Quatre vues de l'apparence de l'enfant d'Ajax aux instants 1, 10, 16 et 25 (26 occurrences en total); (***b***) Les histogrammes RGB correspondants; (***c***) Les histogrammes RGB correspondants où chaque couleur (R, G, B) est représentée par un carré dont sa taille est proportionnelle a la valeur de sa fréquence; (***d***) Illustration de la variation du premier axe principal des histogrammes par rapport au temps.*

que la distribution de l'objet dans cet espace unidimensionnel est probablement bimodale (cette distribution peut être approximée par deux segments de droites parallèles).

• **Taille de la vidéo et indexation des objets suivis** Selon les normes, 24 à 32 images sont projetées par seconde. Si on considère que dans chaque image un unique objet est localisé, un film vidéo de 90 minutes comporte environ 129600 objets à indexer. Ce problème d'indexation devient ingérable en terme de stockage physique et de temps de recherche lorsque les descripteurs extraits des ces objets sont de grandes dimensions. Par exemple, Schiele dans sa thèse [108] utilise des histogrammes multidimensionnels à champs réceptifs de 10^9 cellules (histogramme à six axes et d'une résolution de 32 cellules par axe) et chaque cellule est codée sur 64 bits.

Les solutions proposées dans la littérature pour palier à ce problème sont basées princi-

palement sur la réduction du nombre des images de chaque plan en représentant l'ensemble de ces images par des images clés [92]. Le choix des images clés les plus représentatives du contenu du plan est un problème qui sera abordé dans le chapitre 6 de ce mémoire [58]. Souvent l'image médiane est considérée comme image clé du plan. Selon cette approche, seule l'occurrence médiane d'un objet suivi dans le plan est indexée. En conséquence, la taille de la base d'objets à indexer est divisée par un facteur 129 si on suppose qu'en moyenne un film vidéo est divisé en 1000 plans. Généralement, le nombre de plans d'un film vidéo varie entre 500 et 1000 plans [70] [1]. Notons que pour les films d'action le nombre de plan est plus élevé car la durée des plans est très courte (environ une seconde). Par exemple, le film *Octobre* de S.M. Eisenstein est décomposé d'environ 3225 plans [68].

Cette solution d'images clés peut être envisageable pour le problème d'indexation des objets vidéo posé ici si ces objets sont statiques et si leurs variations intra-plan dans l'espace de descripteurs sont négligeables. Mais malheureusement la plupart des objets d'intérêts des films vidéo sont mobiles et leur variabilité intra-plan dans l'espace de descripteurs est significative (voir figure 3.2). Une étude expérimentale menée sur une séquence vidéo réelle (qui sera détaillé dans le chapitre 4) a montré l'inefficacité de cette méthode pour la reconnaissance d'objets vidéo [55] [57].

3.1.2 La solution proposée

Afin de résoudre les deux problèmes exposés ci-dessus, nous proposons de *modéliser* la variabilité d'un objet suivi dans l'espace de descripteurs de couleurs par une fonction statistique de *mélange de lois*, qui **capture** les différentes apparences intra-plan de l'objet. L'idée de base est de classifier les occurrences semblables en terme d'apparence (dans l'espace de descripteurs) en des groupes. Ensuite chaque groupe construit sera représenté par ses *paramètres statistiques*, ce qui **réduit** considérablement le nombre des descripteurs à indexer par objet suivi. Rappelons que chaque occurrence d'un objet suivi est supposé être représentée par un seul point multidimensionnels dans l'espace de descripteurs. La seule hypothèse que nous fixons dans la suite de ce travail est que la forme de la distribution dans l'espace de descripteurs est capturée par un mélange de gaussiennes, hypothèse que nous discuterons plus tard. Les changements d'apparence de l'objet suivi seront alors modélisés par une *densité de mélange gaussien* dont le nombre des composantes dépend de la complexité de ces changements.

3.1.3 Organisation du chapitre

Dans une première partie nous discuterons les différents descripteurs que nous pourrons choisir pour décrire les apparences intra-plan des objets suivis. Puis notre approche de modélisation de la variabilité intra-plan sera présentée en détail. Dans la section 3.7 quelques résultats expérimentaux d'objets suivis modélisés sont illustrés. Les commentaires et les conclusions d'une telle approche sont discutés dans les deux dernières sections de ce chapitre.

3.2 Représentation de bas niveau des objets : un état de l'art

Dans le cadre de ce travail on se limite à la définition suivante d'un descripteur (définition 3.2.1), mais selon le prochain norme MPEG-7 un descripteur peut avoir d'autres formes, par exemple l'URL de l'image, le texte détecté dans les images d'un film vidéo, etc.

Définition 3.2.1 *Un descripteur est un vecteur de réels, mono- ou multidimensionnels, qui résume d'une manière efficace une ou plusieurs caractéristiques du contenu de l'image (comme par exemple, la couleur, la forme et/ou la texture).*

L'extraction des descripteurs de bas niveau, des images ou des régions d'images, est l'étape de base pour tout système de recherche d'images par le contenu (CBIR[1]). Un descripteur est associé à chaque image (objet) indexé dans la base d'images (d'objets). La recherche d'une image requête consiste premièrement à extraire un descripteur de cette image et ensuite de le comparer avec ceux de la base. Les images de la base les plus proches de l'image requête, en terme de distance de similarité, sont sélectionnées, présentées et ensuite elles peuvent être parcourues par l'utilisateur final du système CBIR.

Dans la suite, quelques propriétés fondamentales des descripteurs sont rappelées. Puis une courte présentation de l'état de l'art des catégories de descripteurs intégrés dans des systèmes de recherche d'images et des systèmes de vidéothèque est faite.

3.2.1 Propriétés d'un descripteur

Dans le domaine de l'indexation d'une base d'images fixes ou d'objets, quelques propriétés de descripteurs sont données pour pouvoir étudier leurs performances en pratiques [94] [63]. Essentiellement, un descripteur doit posséder les trois propriétés suivantes :

1^0) *Discriminant* pour mieux identifier les images similaires et rejeter les images différentes. Si *f(I)* est le descripteur de l'image I, $\mid f(I) - f(I') \mid$ doit être suffisamment grand dès que I et I' ne sont pas similaires (ici $\mid . \mid$ est une distance).

2^0) *Complexité faible* en terme de temps de calcul. Le calcul de $f(.)$ doit être rapide.

3^0) *Taille raisonnable* pour gérer facilement une large base d'images et pour accélérer la procédure de comparaison des descripteurs. La taille du descripteur et le temps de comparaison sont deux facteurs proportionnels.

Une image (objet) indexée et/ou recherchée dans une base d'images (objets) peut être présente sous une variété de prises de vues, de changements d'éclairages, de rotations et d'occultation partielles de son contenu. Sous ces conditions de changements la puissance de discrimination d'un descripteur est fortement liée à son degré d'*invariance* ou de *robustesse*.

1. Content Based Image (and Video) Retrieval

Un descripteur est dit *invariant* à tel type de changement d'image lorsque sa valeur est la même avant et après ce changement : $f(I) = f(t(I))$ où t représente la transformation que l'image I a subite (plus d'informations sur la théorie d'invariance dans [86]). Dans le cas ou la valeur du descripteur a peu varié après ce changement, le descripteur est dit *robuste*; cette robustesse est surtout essentielle en cas de bruit non modélisable comme l'occultation partielle.

3.2.2 Descripteurs pour l'appariement d'images

La première différence entre les systèmes traditionnels de base de données et les base d'images (et de vidéo) est la façon avec laquelle les requêtes sont exprimées. Tandis que les langages d'interrogation standards comme SQL (*structured query language*) peuvent efficacement exprimer l'intention de l'utilisateur, il est très difficile qu'un utilisateur exprime le contenu d'une image. Les requêtes tendent à être exprimées dans des langages naturels et sont elles-mêmes tout à fait complexes. Face à ce problème, la communauté de vision par ordinateur et de reconnaissance de formes a développé des méthodes pour convertir la spécification de requête en un jeu significatif de descripteurs de requête (*Query features*), qui peuvent être appliqués pour rechercher les données visuelles.

Les utilisateurs d'un tel système ne peuvent pas comprendre les méthodes complexes employées pour mettre en application la recherche d'images. Par conséquent une interface plus intuitive doit être donnée aux utilisateurs, de telle sorte que la requête puisse alors être traduite en des paramètres appropriés du système de recherche d'images.

Tout ce qui précède mène au besoin d'une couche intermédiaire entre la requête utilisateur et le processus de recherche qui transmet la requête par l'utilisation d'un descripteur approprié pour interroger l'image. Ces descripteurs devraient être calculés en temps réel et également être capables de capturer l'essence de la requête humaine.

Plusieurs systèmes multimédia développés ces dernières années sont décrits dans la littérature [48] [3]. Parmi ces derniers, les systèmes performants incluent le projet QBIC (*Query By Image Content*) [91], le système Photobook [94], le système MMIS (*Manchester Multimedia Information System*) [47], VisualGREP [76], le projet IDL (*Informedia Digital Library*) [115], SurfImage [90] et celui développé dans notre laboratoire [110].

Par exemple, le système QBIC développé par le centre de recherche *Almaden* de *IBM* utilise une variété de descripteurs pour rechercher les images dans une bases d'images fixes [91]. Il permet à l'utilisateur de rechercher une image requête, que ce soit une image appartenant à la base d'images ou une image d'extérieur ou dessiner manuellement la forme recherchée (*Query-by-Example*), de visualiser et de parcourir (*browse & navigate*) les résultats de la requête (un ensemble d'images). Des descripteurs visuels telles la couleur et la texture sont précalculés et indexés pour toutes les images de la base. Une extension de ce système pour interroger une base vidéo à récemment été rajoutée.

Le système VideoPrep développé dans le cadre de ce travail sur un contrat entre l'INRIA (projet MOVI de l'unité Inria Rhône-Alpes et VISTA de l'unité Inria Rennes) et le centre de recherche d'Alcatel CRC, et destiné à la création de la vidéo hyperliée, utilise des descripteurs globaux de la couleur et des descripteurs locaux de niveaux de gris. Il permet à l'utilisateur de sélectionner un objet d'intérêt dans le film vidéo, de chercher toutes ces

apparitions dans le film, de naviguer et de visualiser les résultats (plus de détails dans le chapitre 8).

Les méthodes d'appariement d'images et d'objets décrites dans la littérature utilisent trois types de descripteurs extraits à partir de l'image. Ces descripteurs sont fondés principalement sur la couleur, la forme et la texture. Quelques systèmes d'indexation, comme décrits ci-dessus, emploient un ou plusieurs descripteurs pour l'appariement des images.

3.2.2.1 Descripteurs globaux de couleur

La couleur est la caractéristique la plus exploitée par les systèmes d'indexation et de recherche d'images par le contenu. Swain et Ballard [120] ont proposé d'utiliser l'histogramme pour décrire la distribution globale des couleurs d'une image.

Un histogramme est un outil statistique qui code une population sous la forme des fréquences (ou effectifs) des individus. Lorsque une image est de haute résolution (16.7 millions de couleurs) une quantification de couleurs est généralement effectuée afin de réduire le nombre de cellules (classes de couleurs) de l'histogramme. La méthode de quantification la plus simple consiste à fusionner les classes de couleurs situées dans un intervalle de couleur donnée. Une autre consiste à appliquer un algorithme de clustering pour fabriquer les classes de couleurs d'une image. Le choix du nombre de cellules est un point critique pour l'appariement d'images par la méthode d' histogramme. Les expérimentations menées dans [108] et [45] montrent que les histogrammes de dimensions réduites fournissent des bon résultats d'appariements.

Les motivations principales d'utilisation de l'histogramme sont la rapidité de son calcul et son indépendance à plusieurs changements d'images: position relative des objets dans l'image, rotation 2D des objets et le changement d'échelle. Également, le calcul de l'histogramme dans un espace de couleurs normalisées, comme par exemple l'espace de chrominances (rgb), le rend invariant aux changements d'éclairages [116] [36]. La transformation de l'espace de couleur RGB en rgb normalisé est donnée par:

$$\begin{cases} r(R,G,B)=\frac{R}{R+G+B} \\ g(R,G,B)=\frac{G}{R+G+B} \\ b(R,G,B)=\frac{B}{R+G+B} \end{cases} \tag{3.1}$$

Cette normalisation de la longueur de chaque couleur RGB (donne $R+G+B=1$) est une méthode performante pour limiter la dépendance de l'intensité [36].

Pour apparier une image requête avec celles de la base d'images, Swain et Ballard calculent les histogrammes dans l'espace de couleurs HSV (équation 3.2), puis utilisent l'intersection d'histogrammes pour les comparer. On a:

$$\begin{cases} H(R,G,B)=\arctan(\sqrt{\frac{\sqrt{3}(G-B)}{(R-G)+(R-B)}}) \\ S(R,G,B)=1-\frac{min(R,G,B)}{I(R,G,B)} \\ V(R,G,B)=\frac{R+G+B}{3} \end{cases} \tag{3.2}$$

où H représente la teinte qui est la caractéristique la plus importante de la couleur, S représente la saturation qui mesure le contenu relatif du blanc d'un couleur (i.e. la

saturation du blanc est égale à zéro, la saturation d'une couleur pure est égale à l'unité et la saturation du noir est indéfinie), et V représente la luminance ou l'intensité définie par la valeur portée par l'axe achromatique du cylindre de MUNSELL.

Notons qu'il n'y a pas des critères théoriques permettant de sélectionner la distance la plus appropriée pour comparer des histogrammes. Schiele dans sa thèse ([108], chapitre 4) a comparé plusieurs distances. Il conclut, sur une base expérimentale d'objets rigides, que la distance du χ^2 normalisée fournie les meilleurs résultats d'appariements :

$$\chi^2(Q,V) = \sum_i \frac{(q_i - v_i)^2}{(q_i + v_i)} \tag{3.3}$$

où q_i et v_i représentent les fréquences de la i^{eme} cellule des histogrammes Q et V respectivement.

Plusieurs améliorations ont été apportées aux histogrammes, soit pour qu'ils soient moins sensibles aux changements d'éclairage, soit pour qu'ils incorporent une certaine information géométrique. Funt et Finlayson [39] ont proposé d'utiliser les dérivées de logarithmes des canaux afin de fournir des caractéristiques invariantes aux changements d'éclairage. Leur objectif est de limiter les effets de la luminosité en normalisant les images vers une illumination standard. Healey et Slater [113] ont aussi proposé des algorithmes du même genre. Ils calculent des invariants de moment de l'histogramme de couleurs en entier. Ces invariants sont fondés sur le modèle linéaire de dimension finie de couleurs permettant la modélisation des changements de l'intensité d'éclairage par une transformation linéaire d'histogrammes.

D'un autre coté, plusieurs auteurs ont proposé d'incorporer des informations spatiales avec la couleur [117] [101] [93]. La plupart des méthodes proposées divisent l'image en régions. Une récente méthode nommée CCV (color coherent vector) [93] utilise une autre approche (raffinement de l'histogramme) : les cellules de l'histogramme sont partitionnées en fonction de la cohérence spatiale des pixels; un pixel étant cohérent s'il appartient à une région d'assez grande taille et uniformément colorée, il est incohérent sinon. Un CCV représente une classification pour chaque couleur dans l'image. Il est facilement implémentable et donne de bien meilleurs résultats que les histogrammes. Par contre, il est sensible aux occultations partielles et aux changements d'illuminance.

Par ailleurs, une autre méthode, basée ni sur le partitionnement de l'image en régions, ni sur un raffinement des histogrammes, et qui semble donner de très bons résultats, a été proposée : il s'agit des corrélogrammes [63]. A cheval entre les méthodes purement locales comme les points d'intérêts et celles purement globales comme la distribution des couleurs, les corrélogrammes tiennent compte aussi bien de la corrélation spatiale des couleurs que de la distribution globale de cette corrélation. Les corrélogrammes apparaissent ainsi comme une solution efficace pour la recherche d'images par le contenu dans une grande base d'images. Selon les concepteurs de cette méthode, une évidence expérimentale suggère que cette nouvelle signature dépasse non seulement les performances de la méthode des histogrammes de couleurs mais aussi les méthodes récentes proposées pour l'affiner et l'améliorer dans le cadre de l'indexation et de la recherche d'images. Un corrélogramme de couleur décrit comment la distribution de probabilité conditionnelle de paires de cou-

leurs change avec la distance. Il est défini comme étant une table indexée par paires de couleurs telle que la k^{eme} entrée pour (i, j) représente la probabilité de trouver un pixel de couleur j à une distance k à partir d'un pixel de couleur i dans l'image. La méthode par corrélogrammes incorpore d'une part la corrélation spatiale des couleurs et décrit d'autre part la distribution globale de la corrélation spatiale locale des couleurs.

Par contre, les corrélogrammes ont des tailles importantes (le minimum conseillé est de 320 pour une distance spatiale d qui vaut 5 et une palette de couleurs m de $4 \times 4 \times 4$). Ils sont aussi sensibles aux occultations partielles et aux variations spatiales du contenu de la scène (déplacement des objets d'une image). D'autre part, le calcul d'un corrélogramme est très lent à cause du balayage de l'image pour les distances spatiales de 1 à d. Les auteurs de ce descripteur proposent d'utiliser la programmation dynamique pour accélérer son calcul. Ils proposent également l'autocorrélogramme qui est comme le corrélogramme dans lequel on ne considère que des paires de couleurs identiques. Son temps de calcul est relativement raisonnable. Par contre, il est moins informatif et donc discriminant que le corrélogramme.

Un autre type de recherche propose de ne pas se baser uniquement sur la couleur mais de faire des histogrammes d'autres caractéristiques locales. Construire un histogramme des résultats de filtres spatiaux orientés ou non est un exemple. Bernt Schiele a proposé une étude qui semble très intéressante à ce sujet [108]: il construit un histogramme multidimensionnels de champs réceptifs et sa méthode semble donner de bons résultats. Cependant ce descripteur a une dimension trop grande et cela représente un handicap majeur pour leur utilisation dans un système d'indexation d'une large base d'images.

Plus récemment encore, une étude de Gevers et Smeulders, qui semble prometteuse, propose une manière efficace de coupler la distribution des couleurs et les invariants géométriques [116] [46]. Dans leur système PicToSeek, les auteurs utilisent différents modèles d'illuminations, la saturation de la couleur, la transition de la couleur, le fond de l'image, un invariant géométrique basé sur le birapport et des histogrammes autour des points d'intérêts détectés dans l'image contour. La transition de la couleur se rapporte au nombre de changements de tonalité (hue) de l'image. Une image avec beaucoup de détails aura plus de transitions qu'une image avec peu de changements. Le fond est supposé être celui de l'image. C'est la cellule de l'histogramme ayant la valeur la plus élevée. L'espace de couleur HSV est choisi car il contient le paramètre V (intensité) représentant l'intensité dans l'image, et pouvant être étendu à la luminosité dans la scène. Un modèle de réflexion pour l'objet est établi et des invariants photométriques de la couleurs sont calculés. Comme nous le verrons dans les expérimentations (chapitres suivants), le modèle $l_1 l_2 l_3$ défini par l'équation 3.4 ([43]) est adopté dans la génération des descripteurs de test.

$$\begin{cases} l_1(R,G,B) = \frac{(R-G)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \\ l_2(R,G,B) = \frac{(R-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \\ l_3(R,G,B) = \frac{(G-B)^2}{(R-G)^2+(R-B)^2+(G-B)^2} \end{cases} \tag{3.4}$$

Le système PicToSeek est décrit en grand détail dans [44]. La base de test utilisée est fabriquée par les auteurs de ce système. Les objets sont acquis par un appareil photo de haute qualité et dans des conditions d'éclairage bien choisies. Il paraît cependant que les

résultats d'appariements varient largement d'un descripteur à un autre, ils sont mauvais lorsque des rotations des objets 3D dans les images requêtes sont présentes.

3.2.2.2 Descripteurs de la forme

La caractéristique la plus importante pour la forme est son contour (2D) ou son enveloppe convexe (3D). Différentes approches d'appariement de la forme sont adoptées par les systèmes de recherche d'images par le contenu. Jain et Vailaya [67] proposent d'apparier la forme sur la base d'un histogramme de direction du gradient aux points de contour de l'image. Le filtre de Canny [19] est appliqué pour obtenir l'image contour. Les histogrammes sont comparés en utilisant l'intersection d'histogramme comme distance. Un histogramme de direction de gradient est invariant à la translation. Il est également invariant aux changements d'échelles s'il est normalisé par le nombre de points du contour. Cette méthode est rapide mais trop sensible aux bruits du signal, aux occultations partielles et aux changements d'éclairages qui affectent beaucoup la détection du contour. Rui et al. [106] utilisent les descripteurs de Fourier pour décrire la forme. Par contre, la transforme de Fourier discrète n'est pas invariante aux transformations affines. Les auteurs définissent un descripteur modifié de Fourier qui est une forme interpolée des coefficients de basse fréquence de descripteurs de Fourier normalisés. L'appariement des images s'effectues en utilisant plusieurs distances comme la distance euclidienne et la distance de Hausdorff. D'autre méthodes basées sur une approximation du contour par des fonctions mathématiques précises tels les B-Splines, sont aussi employées [37] [26]. Pour plus de détails sur la caractérisation de la forme, les lecteurs peuvent se reporter à l'article de Loncaric [77].

3.2.2.3 Descripteurs de la texture

Les caractéristiques visuelles des régions homogènes des images réelles sont souvent identifiées comme texture. Les moments du deuxième ordre, l'énergie, l'entropie, la corrélation, l'homogénéité locale, l'inertie et le contraste, dérivés de la matrice de co-occurrences de niveaux de gris, sont les plus connus pour décrire la texture. Une matrice de co-occurrences est un histogramme à quatre dimensions: $S = f(i, j, d, \theta)$ où i et j sont les niveaux de gris des pixels à une distance d, et θ est l'angle formé par la ligne qui rejoint les deux pixels et l'axe horizontal. Cependant, le calcul de la matrice de co-occurrences est très coûteux surtout lorsque la distance spatiale d est relativement élevée. Notons que le corrélogramme est une extension de la matrice de co-occurrences pour les images couleurs. Lorsque l'orientation est prise en compte dans le calcul du corrélogramme, ce dernier est nommé "corrélogramme d'orientation". Connors et Harlow [27] ont montré sur des exemples réels que ces six mesures de la texture ne sont pas suffisamment discriminantes ([88] chapitre 5). Leur performance de discrimination augmente quand plusieurs valeurs de d sont utilisées. D'autre descripteurs comme les coefficients produits par le filtre de Gabor et les Ondelettes sont aussi utilisés pour décrire la texture [78]. Ils supposent que la texture des régions soit localement homogène. Cependant ces descripteurs ne sont pas invariants à la plupart des changements d'images. Notons que le système Photobook dû à

Pentland et al. [94] est quasiment le seul système de recherche d'image par le contenu qui est basé sur la texture. Les auteurs caractérisent une texture, à l'aide de la décomposition de Wold, par trois mesures de la périodicité, de la directionnalité et de l'aspect aléatoire (*Randomness*).

3.3 Choix de la base de descripteurs : nos données

Les descripteurs globaux discutés ci-dessus sont nullement complets. D'autres catégories de descripteurs existent : les descripteurs locaux par exemple. Les images sont caractérisées par des descripteurs locaux du signal RGB, tels qu'ils ont été définis par Gros et al. [49], lorsqu'ils ont étendus le travail de Schmid et Mohr [110]. Ceux-ci sont des vecteurs de dimension 24 et ils représentent des mesures locales invariantes aux transformations géométriques (rotation, translation, changement d'échelle) et aux changements de la luminosité. Ces mesures sont calculées autour des "point d'intérêt" obtenus par le détecteur de Harris [59].

L'étude de la variabilité présentée ici n'est pas étendue pour les descripteurs locaux mentionés ci-dessus. Une apparence d'objet suivi est représentée par plusieurs vecteurs descripteurs locaux calculés autour des points d'intérêts, entre deux apparences successives la mise en correspondance entre les points d'intérêts est une tâche difficile (objet mobile) et demande un suivi des points d'intérêts pour pouvoir étudier leur variabilité. De plus, ces descripteurs locaux sont bons pour la mise en correspondance mais pas pour l'indexation d'objets en mouvement complexe: des expérimentations que nous avons menées sur une base d'objets vidéo ont montré une performance très faible de ces descripteurs [18].

Les descripteurs globaux proposés jusqu'à ce jour manquent encore de robustesse aux différents changements d'images, et en particulier aux bruits et aux occultations partielles. Leur performance se dégradent rapidement lorsque il s'agit d'apparier des objets réels en mouvement en utilisant une méthode d'appariement classique (voir section 4.5). Ceci est du principalement à l'acquisition des objets non-rigides dans un environnement bruité et à leur apparences variables à travers le temps (voir figure 3.1).

Les insuffisances que nous venons de signaler expliquent que l'extraction de descripteurs reste un domaine de recherche très actif. Aussi, lorsqu'il s'agit de valider une nouvelle approche de reconnaissance ou d'indexation d'images, le choix de la base de descripteurs paraît comme une occupation prioritaire de certains chercheurs.

L'approche retenue dans ce chapitre pour capturer la variabilité intra-plan des objets suivis ainsi que nos approches de classifications de ces objets sont adaptées et expérimentées uniquement pour une base de descripteurs globaux.

Le descripteur global le plus utilisé est l'histogramme de couleur [120]. Rappelons que l'histogramme est invariant aux changements l'échelles, de translation, de l'orientation de la scène et de rotation des objets dans l'image. Aussi, après un certain nombre de traitements élémentaires de normalisation de couleurs (voir section 3.2) l'histogramme est rendu robuste à quelques types de changements d'éclairages. Par contre l'histogramme de couleurs est sensible aux occultations partielles, de même que tout les descripteurs globaux, et peut-être un peu moins que les descripteurs qui intègrent une information spatiale (les

corrélogrammes par exemple). Un défaut de l'histogramme est que deux images différentes ayant la même distribution de la couleurs auraient des histogrammes similaires, et donc les deux images sont appariées semblables. Cependant, nous considérons que cette hypothèse à une probabilité très faible.

Malgré les défauts de l'histogramme, nous nous sommes contentés de le prendre comme un descripteur de base pour valider nos approches. Mais d'autres descripteurs tels que les corrélogrammes peuvent être envisagés.

La base de descripteurs utilisée est composée principalement des histogrammes calculés dans différents espaces de couleurs et de niveaux de gris. Ces espaces sont : RGB, HSV, H, S, I, rgb et $l_1l_2l_3$. Rappelons que les espaces rgb, H et $l_1l_2l_3$ sont invariants aux changements de luminance. Le choix de cette base est motivé par les propriétés intéressantes des histogrammes (voir section précédente), la nature non-rigide des objets exploités (difficulté d'adapter par exemple des descripteurs géométriques) et comme nous le verrons dans la suite le besoin des descripteurs de dimension réduite pour être capable d'estimer d'une manière stable la loi de mélange lorsque le nombre d'apparences intra-plan d'un objet suivi est limité. La réduction de la dimension des descripteurs est réalisée par une analyse en composante principale (section suivante).

3.4 Réduction de la complexité des données

Dans ce travail, et comme nous le verrons dans la suite, la dimension des descripteurs est un paramètre important à gérer. Lorsque un objet suivi possède un petit nombre d'apparences, la modélisation de sa variabilité dans un espace de descripteurs de grande dimension devient instable. Une réduction de la dimension de l'espace par une analyse en composantes principales (ACP) est un pré-traitement que nous appliquerons sur les données : tous les vecteurs descripteurs d'une base d'objets vidéo.

L'ACP est une technique de représentation des données, ayant un caractère optimal selon certains critères algébriques et géométriques, et qu'on utilise sans référence à des hypothèses de nature statistique ni à un modèle particulier [75]. Le but est d'étudier les modes de variations les plus importants, et qui caractérisent au mieux un nuage de points.

Cette technique est souvent utilisée par les approches de classification de visages [121]; une ACP est appliquée sur chaque classe de visages, un visage requête est projeté dans tous ces espaces et ensuite affecté à la classe la plus proche en terme de distance. La recherche est ainsi plus rapide que dans les espaces initiaux (moins de comparaison entre deux vecteurs descripteurs).

Principe Soit un tableau de données $Y = (y_{ij})$ formé de n lignes et de d colonnes. Les lignes représentent les individus (descripteur d'une apparence d'objet) et les colonnes représentent les variables (dimension d'un vecteur descripteur). Les lignes peuvent être considérées comme des réalisations indépendantes de vecteurs aléatoires de l'espace $\mathbb{R}^d$. La distance entre 2 individus a une interprétation géométrique directe. Il s'agit ici de la distance euclidienne classique entre deux points de $\mathbb{R}^d$. Deux individus sont proches (ou voisins) si et seulement si leurs d coordonnées sont proches. Deux individus proches ont

un même comportement vis-à-vis des variables considérées, et on peut les mettre dans une même groupe. Considérons maintenant l'espace $\mathbb{R}^n$ des d variables. Si toutes les n coordonnées de deux variables sont proches, les variables seront représentées par deux points voisins dans $\mathbb{R}^n$. Cela veut dire que ces variables mesurent une même chose ou encore, qu'elles sont corrélées.

Analyse du nuage des individus Globalement, le nuage de points est approché par un hyper ellipsoïde. Le critère algébrique utilisé est que la somme des carrés des distances entre tous les couples d'individus soit maximale.

Les axes de l'ellipsoïde sont les directions des composantes principales et les longueurs des diamètres caractérisent la dispersion du nuage sur chaque composante. On les appelle valeurs propres de la matrice de covariance du nuage, qui, pour le tableau normalisé (centré-réduit), est donnée par : $C = (c_{jk})$ où $c_{jk} = \sum_{i=1}^{n} y_{ij} y_{ik} = cor(j,k) \leq 1$ représente le coefficient de corrélation empirique entre les variables j et k.

Les valeurs propres λ_j sont les termes situés sur la diagonale principale de la matrice obtenue par la diagonalisation de la matrice de corrélation : $D = (d_{jk})_{j,k=1..p}$ $d_{ij} = \lambda_j$ et $d_{jk} = 0$ si $j \neq k$. Les deux matrices sont liées par la relation : $C = P^T D P$ où P est la matrice dont les colonnes sont les vecteurs propres de la matrice de corrélations : $P = (u_1, \ldots, u_p)$. La projection des individus sur le sous espace factoriel E_k déterminé par k vecteurs propres liés aux plus grandes valeurs propres (composantes principales) forme un nuage de points, situé à l'intérieur de l'ellipse d'intersection entre le sous-espace E_k et l'hyper ellipsoïde.

Test de validité La qualité globale de représentation des données initiales sur le sous-espace factoriel E_k est mesurée par le pourcentage d'inertie (la variance) pris en compte par E_k :

$$Q_{E_k} = \frac{\sum_{j \in E_k} \lambda_j}{\sum_{i=1}^{d} \lambda_i} \times 100. \tag{3.5}$$

Les inerties associées aux λ_j peuvent être aussi représentées sur un histogramme. Si la décroissance est régulière, alors les données sont peu structurées et les variables faiblement corrélées, voir linéairement indépendantes. L'analyse ne fournira pas des résultats intéressants. Si, au contraire, la décroissance est assez irrégulière, présentant des paliers, visibles surtout sur les premières valeurs, alors il y a des fortes corrélations entre les variables et on peut réduire de beaucoup le nombre de composantes significatives. Un autre critère empirique nous permet de s'arrêter là où on trouve un coude sur l'histogramme.

Défauts de l'ACP La projection de données se fait d'une manière linéaire. Si ces données ont des structures complexes le risque de confusion augmente. En plus, le critère des moindres carrés qui est à la base de cette analyse suppose implicitement que les

données ont une distribution gaussienne et si ce n'est pas le cas, les valeurs aberrantes ont une contribution significative à l'inertie des axes ce qui fausse les résultats de l'analyse. Un recours à des méthodes de pré-traitement des données pour éliminer les valeurs aberrantes est très utile (analyse de rangs par exemple). D'autres méthodes d'analyse de données non-linéaires comme par exemple l'analyse en composante curvilignes (ACC) [30] et la méthode *multidimensionnel scaling* (MDS) [73] peuvent être employées. Druga [33] dans son travail de DEA à l'équipe MOVI, a montré que l'ACC ne donne pas meilleurs résultats que l'ACP. Elle explique ceci par l'inexistence d'une forte structure dans les données (descripteurs d'invariants différentielles [49] [109]) et/ou par des relations linéaires ou linéarisables entre les composantes. Par contre, la méthode MDS est beaucoup moins utilisée car elle est trop coûteuse en terme de calcul.

3.5 Variabilité intra-plan des objets suivis : le problème

Avant de proposer une stratégie de classification des objets suivis, qu'elle soit supervisée ou automatique – objet des chapitres suivants –, il s'avère très utile de proposer une technique de caractérisation de bas-niveau, bien adaptée à nos données visuelles : les objets mobiles. La suite de ce chapitre se focalise sur ce point fondamental de notre travail.

Un objet suivi est une suite continue d'apparences de cet objet à l'intérieur des images du plan vidéo (voir le chapitre précèdent pour plus de détails sur les techniques utilisées pour la localisation et le suivi des objets mobiles). L'objet suivi à un instant t dans le plan vidéo sera désigné par *apparence* intra-plan dans la suite. Nous supposons dans la suite que chaque apparence est représentée par un point unique dans l'espace de descripteurs. Plus formellement, on extrait de l'apparence i de l'objet suivi un descripteur; ce descripteur est décrit sous la forme d'un vecteur v_i de d valeurs réelles; un espace de $\mathbb{R}^d$ est ainsi défini. Une apparence est représentée par un point unique $y_i \in \mathbb{R}^d$. Cette représentation unique d'une apparence dans l'espace de descripteur est valable pour tout descripteur global (voir section 3.2). Les extensions de cette étude pour les descripteurs locaux (les invariants différentiels autour des points d'intérêt, etc.) seront discutées dans les perspectives de ce chapitre.

Si l'objet suivi est mobile et non-rigide – le cas le plus fréquent dans les films vidéo –, plusieurs apparences visuellement différentes peuvent être distinguées pour cet objet. L'apparition progressive intra-plan d'un objet suivi sous une variété de changements d'images (voir l'introduction) produit une **variabilité** dans sa distribution spatiale dans l'espace de descripteurs. Un objet suivi est représenté par un ensemble de descripteurs extraits de ses apparences. Soit $Y = (y_1, ..., y_n)$ la collection de descripteurs (la distribution) de l'objet suivi ayant n apparences, avec y_i le vecteur descripteur de dimension d qui caractérise l'apparence i. La variabilité de l'objet suivi est expliquée par une dispersion des individus y_i dans l'espace $\mathbb{R}^d$. La dispersion est exprimée souvent autour d'un centre. A ce stade on utilise le terme "**degré de variabilité**" pour distinguer entre distribution unimodale et multimodale. La distribution est multimodale si elle est divisible en plusieurs classes différentes. Le degré de variabilité d'une distribution Y dépend du changement significatif de l'apparence intra-plan de l'objet suivi et de la robustesse du descripteur extrait des

apparences aux changement d'images.

Aussi, la variabilité d'une distribution d'un objet suivi à travers le temps pourra être bien illustrée par les points de courbures fortes de sa trajectoire spatio-temporelle [5]. Les points de cette trajectoire sont dans un espace de dimension $d+1$, où la dimension temporelle est rajoutée aux d dimensions spatiales. Les points de courbures de la trajectoire signalent des changements significatifs dans l'apparence de l'objet suivi. Notons que le nombre de points de courbures ne correspond pas du tout au degré de variabilité (voir la définition ci-dessus). Ceci est du au fait que beaucoup de points de la trajectoire spatio-temporelle peuvent correspondre à différents segments de la trajectoire sachant qu'ils appartiennent à la même classe d'apparences de l'objet. L'exemple de la figure 3.2.d, détaillé dans l'introduction de ce chapitre, illustre une trajectoire spatio-temporelle bidimensionnelle de l'enfant suivi. Chaque apparence de l'objet à un instant donné est représentée par l'histogramme de couleurs projeté dans le premier plan factoriel. Un rappel de l'ACP appliquée ici pour réduire la dimension de de l'espace de descripteurs est fait dans la section 3.4. Sur cette trajectoire, on voit nettement un point de courbure forte. En conséquence, deux classes d'apparences de l'enfant peuvent être construites : une classe lorsque l'enfant se place entièrement dans la lumière et une deuxième lorsque l'enfant a partiellement disparu dans la partie sombre.

La détermination automatique du degré de variabilité ou plus couramment, le **nombre de classes** d'une distribution d'un objet suivi, est un point fondamental. Ceci est important pour la phase suivante qui porte sur l'appariement des objets suivis basé sur les modèles d'apparences intra-plan des objets. Si on surestime ou sous-estime le nombre de classes d'apparences des objets suivis, le risque de faux appariements entre ces objets augmente. Nous reviendrons sur ce point ultérieurement et en détail dans les deux chapitres suivants.

L'approche que nous proposons pour caractériser un objet suivi consiste à modéliser la distribution de Y comme une fonction de densité de probabilité jointe, $f(y \mid Y, \theta)$, où θ représente l'ensemble de paramètres du modèle f. Nous supposons que f pourra être estimée comme une fonction de densité de mélange gaussien; nous détaillons ce point dans la section suivante. Cette modélisation capture d'une manière efficace la variabilité multimodale de l'objet suivi dans l'espace de descripteurs. Premièrement, les apparences de l'objet suivi sont regroupées dans des classes d'équivalences homogènes. Dans la suite, ces classes sont nommées **classes d'apparences intra-plan**. Deuxièmement, chaque classe d'apparence est caractérisée par quelques paramètres statistiques (la moyenne et la matrice de covariances dans le cas gaussien) estimés à partir de ses éléments (les individus de cette classe). Les modèles f seront utilisés dans l'appariement inter-plan des objets suivis dans une séquence vidéo. Deux objets suivis sont jugés similaires s'ils ont deux classes d'apparences proches au sens d'une distance donnée. Cela justifie notre approche d'identification des classes d'apparences intra-plan d'un objet suivi. Nous détaillons la technique d'usage des modèles d'apparences intra-plan et leurs comparaisons dans les deux chapitres suivants.

3.6 Modélisation par mélange de lois : notre approche

On présente dans cette section la modélisation statistique du problème de la variabilité évoqué précédemment par un mélange de lois. Dans ce travail, on adopte l'hypothèse que la forme de la distribution de chaque classe d'apparences d'un objet suivi suit une loi gaussienne multivariée. Le modèle du mélange est d'abord rappelé dans sa formulation classique. On détaille ensuite le cas des mélanges gaussiens multivariés. Puis on introduit les modèles gaussiens avec contraintes pour éviter la surestimation et/ou l'instabilité de l'estimation des paramètres du mélange, surtout quand le nombre d'apparences d'un objet suivi est relativement petit par rapport à la dimension de l'espace de descripteurs. Enfin, on décrit les méthodes statistiques pour sélectionner automatiquement la structure d'un modèle de mélange, c'est-à-dire le modèle gaussien et le nombre de composantes (classes) de la densité du mélange. Le nombre de classes ou le degré de variabilité diffère d'un objet suivi à l'autre (voir la section précédente).

3.6.1 Modèle de mélange

En classification automatique, on cherche généralement à détecter un regroupement des observations en J classes homogènes et distinctes les unes des autres. Aussi, lorsqu'on cherche à définir un modèle probabiliste, il paraît naturel de supposer que les observations ont été générées à partir de J distributions homogènes, c'est-à-dire concentrées autour de leur valeur centrale et se chevauchant peu. Les modèles de mélange constituent à cet égard un cadre simple et adapté pour de tels problèmes de classification.

3.6.1.1 Caractérisation d'une distribution mélange

Dans un modèle de mélange les observations $\mathbf{y} = (y_1, ..., y_n)$ sont supposées être des réalisations indépendamment et identiquement distribués (i.i.d.)de n vecteurs aléatoires $\mathbf{Y} = (Y_1, ..., Y_n)$. La distribution du vecteur aléatoire parent est supposée être un mélange de J distributions, caractérisées par des fonctions de densité de probabilité $\varphi(. \mid \alpha_j)$, $1 \leq j \leq J$. Ces J distributions ont pour paramètres $\alpha_1, ..., \alpha_J$, et sont mélangées selon des proportions respectives $p_1, ..., p_J$, avec $0 < p_j < 1$ et $\sum_{j=1}^{J} p_j = 1$. La densité de probabilité en un point $y \in \mathbb{R}^d$ est ainsi donnée par :

$$f(y|\theta) = \sum_{j=1}^{J} p_j \varphi(y|\alpha_j) \tag{3.6}$$

Le vecteur θ dénote les paramètres inconnus du mélange. Dans le cas le plus général, $\theta = (p_1, \ldots, p_J, \alpha_1, \ldots, \alpha_J)$.

3.6.1.2 Mélanges gaussiens

Lorsque les données sont quantitatives, c'est-à-dire à valeurs réelles continues, et en absence de connaissances particulières concernant les distributions du mélange, il est cou-

rant de supposer que chaque classe suit une distribution gaussienne multivariée. Dans cette situation multivariée, la densité de la composante j en un point $y \in \mathbb{R}^d$ est donnée par

$$\varphi(y \mid \mu_j, \mathbf{\Sigma}_j) = \frac{1}{(2\pi)^{d/2} \mid \mathbf{\Sigma}_j \mid^{1/2}} exp\left(-\frac{1}{2}(y-\mu_j)'\mathbf{\Sigma}_j^{-1}(y-\mu_j)\right) \tag{3.7}$$

où $\mid (.) \mid$ dénote le déterminant de la matrice $(.)$. La distribution de chaque classe est paramétrée par son vecteur moyenne μ_j $(d \times 1)$ et sa matrice de variance Σ_j $(d \times d$ symétrique définie positive).

Les paramètres à estimer pour le mélange $f(x \mid \theta)$ sont donc :

$$\theta = (\theta_j) \text{pour } j = 1, \ldots, J \text{ et avec } \theta_j = (p_j, \mu_j, \Sigma_j).$$

La figure 3.3 montre le type d'allure que peuvent prendre des données distribuées selon un modèle de mélange gaussien, respectivement dans $\mathbb{R}$ et $\mathbb{R}^2$. Le figure 3.3.a illustre les données x dans $\mathbb{R}^2$ $(n = 114)$. La figure 3.3.b montre la partition obtenue par l'algorithme EM (voir section 3.6.5) où le nombre de classes (fixé à 5) et le modèle gaussien général (voir section 3.6.2) ont été fixés a priori. La densité du mélange correspondante, dans $\mathbb{R}$ et $\mathbb{R}^2$ est illustrée dans la figure 3.3.c et 3.3.d respectivement.

3.6.2 Modèles gaussiens avec contraintes

3.6.2.1 Motivation pour ce travail

La réduction de la dimension de l'espace de descripteurs par une ACP est une étape fondamentale pour mieux conditionner l'estimation des paramètres du mélange gaussien. Mais la dimension initiale des descripteurs globaux est généralement élevée (64, 128, etc.), et la dimension retenue du nouvel espace de descripteurs après l'ACP, pour une qualité de représentation de 90% par exemple, reste encore grande vis-à-vis du petit nombre d'apparences intra-plan de quelques objets suivis dans la vidéo (cas où la durée du plan vidéo est d'une seconde). On reste alors face à un problème de surparamétrisation. Une solution pour ce problème consiste à réduire la complexité de l'approche de modélisation de la variabilité intra-plan. Ceci pourra être fait par réduction du nombre de paramètres à estimer surtout les coefficients de la matrice de variance Σ. Nous introduisons dans la suite la notion des modèles gaussiens *parcimonieux*. Lorsque l'on modélise la distribution d'un échantillon par un mélange gaussien différentes hypothèses peuvent être adoptées en fonction du problème traité. Si l'on utilise le modèle le plus général, on laisse tous les paramètres varier : le vecteur de paramètres inconnu est θ. Cependant, il est parfois avantageux en pratique de se baser sur un modèle plus contraint, en supposant par exemple que les matrices de variance sont identiques. La section 3.6.2.3 décrit les 7 modèles gaussiens dont nous nous servis.

Avoir moins de paramètres à estimer rend leur estimation plus fiable lorsque l'on dispose de peu d'observations. En principe, l'utilisateur fixe ces contraintes d'après les connaissances dont il dispose sur le problème modélisé. Mais, dans ce travail il s'agit d'appliquer l'approche de modélisation intra-plan sur la distribution de chaque objet suivi séparément. Cette distribution a probablement une structure différente d'un objet suivi à

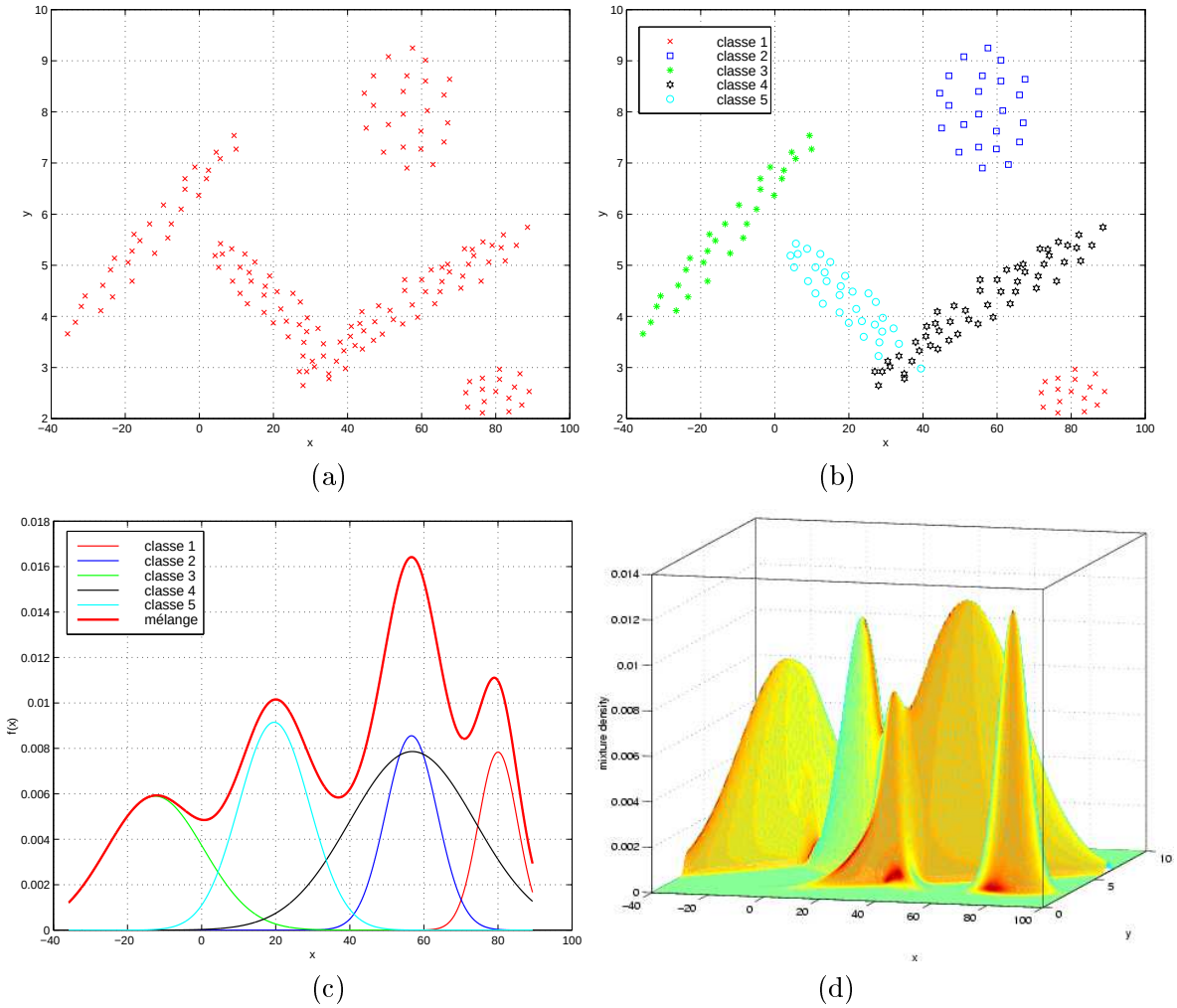


FIG. 3.3: *(a) illustration de données dans $\mathbb{R}^2$; (b) la partition correspondante en cinq classes; (c) la densité du mélange dans $\mathbb{R}$ et (d) dans $\mathbb{R}^2$.*

un autre. Donc, fixer un modèle gaussien avec contraintes pour toutes les distributions des objets suivis, ou bien laisser "l'auteur de la vidéo hyperliée" de la vidéo choisir un modèle manuellement pour chaque distribution, sont deux solutions à éviter. En effet, il s'avère très utile dans ce cas d'utiliser une stratégie qui permette de sélectionner automatiquement le modèle gaussien modélisant le mieux la distribution d'un objet suivi. Pour ce faire, la section 3.6.6 introduit quelques critères implementés dans ce travail.

3.6.2.2 Les 4 modèles gaussiens de bases

Les connaissances a priori les plus classiques dont l'utilisateur dispose portent souvent sur les proportions p_j et les matrices de variances des classes Σ_j. Ainsi, les proportions peuvent être supposées identiques, et n'ont pas besoin d'être estimées ($p_1 = \ldots = p_J =$

$1/J$). Concernant les matrices de variance, quatre modèles sont couramment utilisés (cf. [34] [15]) :

1. Modèle linéaire sphérique: $\Sigma_1 = \ldots = \Sigma_J = \sigma^2\mathbf{I}$, où $\mathbf{I}$ dénote la matrice identité $d \times d$). Dans chaque classe les variables sont supposées indépendantes et dans toutes les classes les variables ont la même variance inconnue $\sigma^2 > 0$.

2. Modèle linéaire diagonal: $\Sigma_1 = \ldots = \Sigma_J = \text{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ avec $(\sigma_1^2, \ldots, \sigma_d^2)$ inconnus représentent les éléments diagonaux de la matrice diagonale $\text{diag}(\sigma_1^2, \ldots, \sigma_d^2)$. Dans chaque classe les variables sont supposées indépendantes.

3. Modèle linéaire général: $\Sigma_1 = \ldots = \Sigma_J = \Sigma$. A l'intérieur de chaque classe, les variables peuvent être corrélées, et les différentes classes partagent la même matrice de variance inconnue Σ (matrice $d \times d$ symétrique définie positive).

4. Modèle quadratique : aucune contrainte sur les Σ_j. Les variables peuvent être corrélées à l'intérieur de chaque classe et les classes peuvent posséder des structure de covariance différente.

Les trois premiers modèles sont dits linéaires car les séparations entre les classes sont des hyper-plans dans $\mathbb{R}^d$. Ceci est dû au fait que, dans ces trois modèles, les classes ont la même matrice de variance.

3.6.2.3 Les modèles parcimonieux

Banfield et Raftery [6] proposent une décomposition spectrale de la matrice de variance afin de raffiner les quatre modèles gaussiens présentés ci-dessus. Cette décomposition permet de spécifier la structure de variance des classes par des paramètres intuitifs, interprétables d'un point de vue aussi bien statistique que géométrique. Nous présentons ici la méthode adoptée dans ce travail, la décomposition de Celeux et Govaert [22]. Cette décomposition modifie un peu celle de Banfield et Raftery en facilitant le calcul des estimateurs du maximum de vraisemblance. L'idée de base est d'écrire chaque matrice de variance sous la forme

$$\Sigma_j = \lambda_j D_j A_j D_j'$$

où

- le paramètre scalaire positif $\lambda_j = \det(\Sigma_j)^{1/d}$, appelé *volume* de la classe j, indique la dispersion de la classe dans $\mathbb{R}^d$; ce paramètre, homogène à la variance des variables de la classe est égal à la moyenne géométrique des valeurs propres de Σ_j;

- la matrice $d \times d$ orthogonale D_j est constituée en colonne des vecteurs propres de Σ_j. Appelée *orientation*, elle représente géométriquement les axes de l'ellipsoïde de dispersion de la classe j, et statistiquement le degré de corrélation entre les variables dans la classe j;

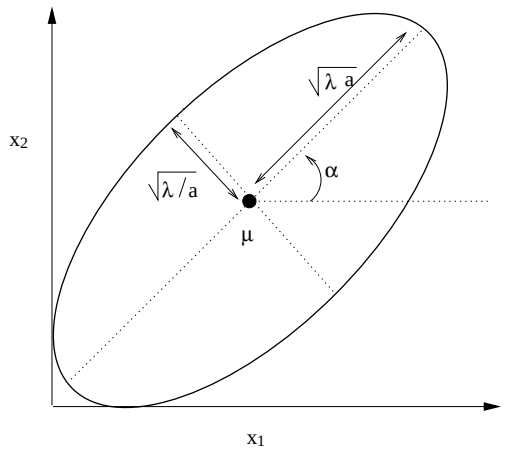


FIG. 3.4: *Décomposition spectrale de la matrice de variance : relation entre l'ellipse de dispersion et les paramètres de volume, de forme et d'orientation.*

– La matrice diagonale $A_j = \text{diag}(a_{j1}, \ldots, a_{jd})$ est formée par les valeurs propres de Σ_j, rangées par ordre décroissant et normalisées de sorte à ce que $\det(A_j) = \prod_{\ell=1}^{d} a_{jd} = 1$. Appelé paramètre de *forme*, A_j indique l'allongement relatif de l'ellipsoïde de dispersion de la classe j le long de ses axes. D'un point de vue statistique, les a_{jd} peuvent être interprétés comme les dispersions relatives des variables après le changement de base induit par la matrice d'orientation D_j.

Selon si ces trois paramètres entre les classes varient ou sont égaux, selon les formes supposées sphériques ($A_j = \mathbf{I}$) ou non et selon si les orientations sont parallèles aux axes (D_j = matrice de permutation de la base canonique) ou obliques, on obtient 14 modèles gaussiens, du plus parcimonieux au moins parcimonieux [22]. Ces 14 modèles peuvent être regroupés en 3 familles :

1. famille sphérique : $A_j = \mathbf{I}$, ce qui donne une matrice $\Sigma_j = \lambda_j \mathbf{I}$ sphérique; l'orientation D_j n'a donc aucune influence dans ce cas.

2. famille diagonale : la matrice de forme A_j est quelconque, et la matrice d'orientation D_j définit une permutation de la base canonique, ce qui donne une matrice diagonale; on note que $B_j = D_j A j D_j'$ la matrice normalisé (diagonale) avec $det(B) = 1$;

3. famille générale : forme et orientation quelconques, la matrice Σ_j n'est pas forcément diagonale; on note $C_j = D_j A_j D_j'$ la matrice normalisée.

La liste complète de ces 14 modèles gaussiens ainsi que l'étape de maximisation M de l'algorithme EM sont détaillées dans l'article de [22]. Dans ce travail, nous nous sommes restreints à la moitié de ces 14 modèles résumés dans le tableau 3.1. Ce choix est motivé principalement par deux choses : (1) ces modèles, partagés entre les trois grandes familles de modèles gaussiens soulignées ci-dessus, ont des complexités variables (faible, moyenne

et forte), ils peuvent être donc adaptés à une distribution d'un objet suivi avec un peu ou beaucoup d'apparences intra-plan; (2) le temps de la sélection automatique de la structure d'un mélange gaussien (section 3.6.6) sera réduit au moins au moitié et donc une modélisation de la variabilité est faisable dans une durée raisonnable.

Famille	Modèles	Nombre de paramètres	
		Cas général	J = 2 et d = 10
Sphérique	$[\lambda I]$	$\alpha + 1$	22
	$[\lambda_j I]$	$\alpha + J$	23
Diagonale	$[\lambda_j B]$	$\alpha + d + J - 1$	32
	$[\lambda_j B_j]$	$\alpha + Jd$	41
Générale	$[\lambda_j C]$	$\alpha + \beta + J - 1$	76
	$[\lambda C_j]$	$\alpha + J\beta - (J - 1)$	130
	$[\lambda_j Cj]$	$\alpha + J\beta$	131

TAB. 3.1: *Liste des 7 modèles gaussiens retenus;* $\beta = (d(d+1)/2)$ *et* $\alpha = Jd + J - 1$.

Dans le tableau 3.1 chaque modèle est désigné par une notation abrégée indiquant quels paramètres sont identiques ou différents entre les classes. Ainsi, dans cette notation, le modèle imposant un même volume entre toutes les classes est noté $[\lambda]$, le modèle permettant des volumes différents entre classes est noté $[\lambda_j]$. Aussi, $[\lambda_j D_j A D_j']$ désigne le modèle à volume et orientation libres entre les classes mais à forme identique. les quatres modèles de variance classiques cités plus haut seraient respectivement désignés par $[\lambda I]$ (modèle linéaire sphérique), $[\lambda I]$ (linéaire diagonal), $[\lambda C]$ (linéaire général), $[\lambda_j C_j]$ (quadratique). Dans [22], on souligne par exemple que le modèle $[\lambda_j I]$, matrices de variances sphériques de volume différents $\Sigma_j = \sigma_j I$, requiert l'estimation de seulement J paramètres pour les variances, contre $J \times d \times (d+1)/2$ paramètres avec le modèle $[\lambda_j C_j]$.

3.6.2.4 Ellipse de dispersion

Pour une distribution gaussienne Y dans $\mathbb{R}^2$ de centre μ et de matrice de variance $\Sigma = \lambda D A D'$, on peut noter que le lieu des points $y \in \mathbb{R}^2$ ayant une densité est caractérisé par une équation de la forme

$$(y - \mu)' \Sigma^{-1} (y - \mu) = \Delta \tag{3.8}$$

où Δ est une constante positive qui dépend du niveau de densité choisi. L'équation 3.8 définit en fait l'ensemble des points y situés à une distance de Mahalanobis de μ égale à Δ, distance pondérée par Σ. On verra plus loin que la courbe ainsi définie contient comme donnée une proportion p_Δ des points issus de la distribution Y.

L'équation 3.8 définit une ellipse centrée en μ. En effet, cette équation s'écrit aussi (D étant orthogonale, $D^{-1} = D'$):

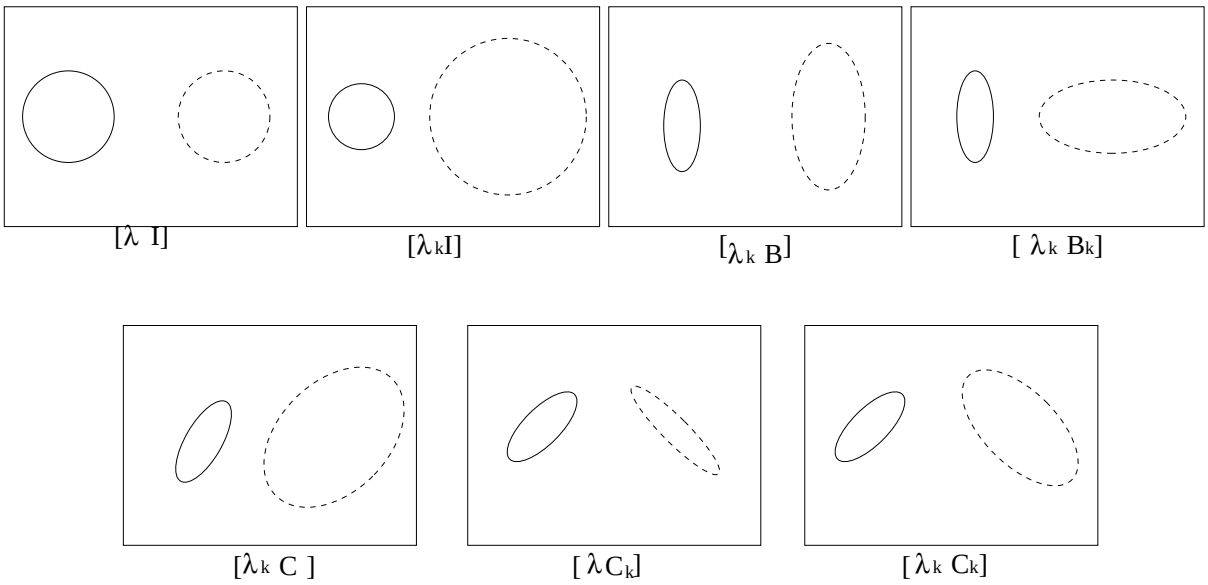


FIG. 3.5: *Illustration des ellipses de dispersion des 7 modèles gaussiens utilisés pour modéliser des données de deux composantes gaussiennes. Chaque couple d'ellipses est associé à un seul modèle gaussien.*

$$[D'D(y-\mu)]'D'(\lambda A)^{-1}D(y-\mu) = \Delta$$

$$[D(y-\mu)]'(\lambda A)^{-1}D(y-\mu) = \Delta.$$

On peut faire le changement de variable

$$\hat{y} \approx D(y-\mu)$$

ce qui revient se placer dans le repère centré en μ et orienté suivant les axes définis par D. Par ailleurs, dans $\mathbb{R}^2$, la matrice de forme A a pour expression

$$A = \begin{pmatrix} a & 0 \\ 0 & 1/a \end{pmatrix}$$

avec $a \in \mathbb{R}$. L'équation ci-dessus peut donc s'écrire

$$\hat{y}' \begin{pmatrix} \frac{1}{\lambda a} & 0 \\ 0 & \frac{a}{\lambda} \end{pmatrix} \hat{y} = \Delta$$

$$\frac{(\hat{y_1})^2}{\lambda a} + \frac{(\hat{y_2})^2}{\frac{\lambda}{a}} = \Delta$$

ce qui définit une ellipse centrée en μ et orientée suivant les axes du nouveau repère. Les grand et petit axes de cette ellipse ont pour longueurs respectives $\sqrt{\Delta\lambda a}$ et $\sqrt{\Delta\frac{\lambda}{a}}$.

3.6.3 Estimation du paramètrage du mélange

L'estimation consiste à donner des valeurs approchées aux paramètres d'un échantillon à l'aide de n observations issues de cet échantillon. Différentes méthodes d'estimation du paramètre du mélange gaussiens multivariés θ peuvent être appliquées: maximum de vraisemblance (MV), estimation bayésienne, théorie du codage, etc. Le paragraphe 3.6.3.2 décrit brièvement le principe d'estimation par MV et l'estimation bayésienne. Nous détaillons ci-après l'estimation par MV, qui sera utilisée dans notre approche d'identification des classes d'apparences intra-plan. Plus précisément, on se focalise sur ce cas automatique où on dispose seulement des observations $y_1, ..., y_n$, les vecteurs descripteurs des apparences d'un objet suivi.

Définition 3.6.1 *Estimateur-Estimation : On appelle estimateur T du paramètre θ toute fonction ϕ de l'échantillon $Y_1, ..., Y_n$, $T = \phi(Y_1, ..., Y_n)$. La réalisation de cet estimateur, $t = \phi(y_1, ..., y_n)$, est appelée estimation et sera notée dans la suite par $\hat{\theta}$.*

3.6.3.1 Conditions d'identifiabilité du mélange

Pour estimer les paramètres d'un modèle -le mélange dans notre cas- celui-ci doit-être *identifiable*. L'indentifiabilité signifie que le modèle, f, possède une décomposition unique en terme de composantes, φ_j. Autrement dit, dans un modèle identifiable, si deux mélanges produisent une même fonction de densité, on a :

$$\sum_{k=1}^{K} p_k \varphi(.|\alpha_k) = \sum_{j=1}^{J} p'_j \varphi(.|\alpha'_j)$$

d'où, compte-tenu de l'unicité de la décomposition :

$$\begin{cases} K = J \\ \text{et} \\ \forall k, p_k = p'_k \text{ et } \varphi(.|\alpha_k) = \varphi(.|\alpha'_k). \end{cases}$$

Yakowitz et Spragins présentent dans [126] une étude approfondie sur la caractérisation des mélanges identifiables. Ils montrent que les mélanges gaussiens sont en particulier identifiables. Au contraire, les mélanges de lois uniformes consituent un cas typique de modèle non identifiable. Pour le constater, prenons l'exemple de la densité uniforme $U_{[0;1]}$ sur l'intervalle $[0, 1]$. Cette densité peut être décomposée en deux densités uniformes du type $U_{[0,p]}$ et $U_{[p,1]}$ en proportions respectives p et $1 - p$, $\forall p \in]0, 1[$.

3.6.3.2 Principes d'estimation

Estimation par maximum de vraisemblance Le principe du maximum de vraisemblance est l'un des plus utilisés pour estimer le paramètre θ d'une distribution f en se basant sur une réalisation y d'un échantillon Y de cette distribution. Cette démarche consiste à chercher l'expression de paramètre θ qui maximise la probabilité $f(y \mid \theta)$ d'observer x avec comme vecteur de paramètres θ. Pour une réalisation y fixée, la probabilité

$f(y \mid \theta)$ est une fonction de paramètre appelée vraisemblance. En pratique, on cherche plutôt à maximiser le logarithme de la vraisemblance, noté $L(\theta)$, ce qui est équivalent et conduit généralement à des calculs plus simples. On cherche donc :

$$\hat{\theta}_{MV}(y) = \arg\max_{\theta} \Big(L(\theta) = \log f(y \mid \theta) \Big)$$

La log-vraisemblance du paramètre θ d'un mélange gaussiens $f(y \mid \theta)$ (équations 3.6, 3.7) est donnée par :

$$L(\theta) = \sum_{i=1}^{n} \log \left\{ \sum_{j=1}^{J} p_j \varphi(y_i | \mu_j, \Sigma_j) \right\}. \tag{3.9}$$

D'autres formulations du MV existent lorsque les données et leurs labels sont partiellement ou complétement connus.

Estimation bayésienne. Dans l'approche bayésienne de l'inférence statistique, l'estimation de paramètre θ d'un modèle est vue comme un problème de *décision* concernant le vrai paramètre inconnu θ au vu de données $y = \{y_1, \ldots, y_n\}$. On se donne pour cela :

- l'estimateur $l(\hat{\theta} \mid \theta)$ qui est une fonction de coût indiquant le coût d'une décision $\hat{\theta}$ si la vrai valeur est θ; $l(\hat{\theta} \mid \theta)$ associe a chaque décision $\hat{\theta} \in \{\text{espace des décisions}\} \times \{\text{espace des paramètres}\}$ une valeur réelle positive;
- et une loi a priori des paramètres $\pi(\theta)$.

L'estimateur de Bayes $\hat{\theta}$ de θ est celui qui minimise le risque a posteriori, donné par

$$T(\hat{\theta} \mid \theta) = E[l(\hat{\theta} \mid \theta) \mid y] = \int l(\hat{\theta} \mid \theta) \pi(\theta \mid y) d\theta$$

avec $\pi(\theta \mid y)$ dénote la loi a posteriori des paramètres qui est déterminée par la connaissance de la loi a priori et de la vraisemblance (formule de Bayes) :

$$\pi(\theta \mid y) = \frac{\pi(\theta)\pi(y \mid \theta)}{\pi(y)} = \frac{\pi(\theta)\pi(y \mid \theta)}{\int \pi(\theta)\pi(y \mid \theta) d\theta}$$

La fonction de coût la plus utilisée est le coût quadratique, $l(\hat{\theta} \mid \theta) = (\hat{\theta} - \theta)'(\hat{\theta} - \theta)$, qui donne généralement lieu à des calculs plus simples. La minimisation du risque a posteriori $T(\hat{\theta} \mid \theta)$ sur $\hat{\theta}$ est résolue explicitement dans ce cas quadratique, mais nécessite la connaissance de la loi a posteriori. Malheureusement, en mélange, la loi a posteriori, quoique explicite, n'est pas calculable pour une taille d'échantillon raisonnable. Ceci est confirmé par le contre exemple traité par Robert [103] (exemple 9.5 du chapitre 9). Récemment, des méthodes de simulations des réalisations, connues sous le nom MCMC (*Markov Chain Monte Carlo*) sont proposées pour palier ce problème de calcul direct de la loi a posteriori. Nous nous servirons des méthodes de Monte Carlo dans l'évaluation de la distance de Kullback calculée entre deux distributions gaussiennes (voir section 5.3).

3.6.4 Maximum de vraisemblance par l'algorithme EM

3.6.4.1 Information manquante pour le mélange

En classification automatique, l'idée de base est de trouver un ensemble des variables manquantes, $z = (z_1, \ldots, z_n)$, qui relient les observations $y = (y_1, \ldots, y_n)$ à J classes inconnues. Les variables manquantes z représentent les labels ou les classes des observations : y_i est classé dans la j^{eme} classe si et seulement si $z_i = j$. Soit $x = (y, z)$ les données complètes. Le principe du maximum de vraisemblance pour ces données complètes x donnerait lieu à des calculs simples en général. En effet, la vraisemblance du θ pour x est alors

$$\begin{aligned} L(x,\theta) = \log f(y,z \mid \theta) &= \sum_{i=1}^{n} \log\Big(p_{z_i}\varphi(y_i,\alpha_{z_i})\Big) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{J} z_{ij}\log\Big(p_j\varphi(y_i,\alpha_j)\Big) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{J} z_{ij}\log p_j + \sum_{i=1}^{n}\underbrace{\sum_{j=1}^{J} z_{ij}\log\varphi(y_i,\alpha_j)}_{\log f(y_i,|\theta)} \end{aligned} \tag{3.10}$$

avec $z_{ij} = 1$ ou 0 selon que $z_i = j$ ou non.

La recherche des paramètres des distributions pourrait se faire en maximisant séparement la vraisemblance $f(y_i, \mid \theta)$ à l'intérieur de chaque classe. Cette maximisation ne pose généralement pas de difficulté dans la mesure où les composants d'un mélange ont la plupart du temps des distributions simples.

L'algorithme *Expectation-Maximization* (EM) est précisement conçu pour ce type de problème d'estimation, où le principe du maximum de vraisemblance pour les données complètes x donnerait des calculs simples, mais où l'on n'observe en fait qu'une partie y de ces données [31].

3.6.4.2 Algorithme itératif

Il s'agit d'une procédure d'estimation itérative, qui commence avec une valeur initiale des paramètres θ^0. Chaque itération consiste ensuite à calculer les nouveaux paramètres θ^{m+1} à partir de ceux de l'itération précédente θ^m de façcon à maximiser une fonction notée $Q(\theta, \theta^m)$, ainsi définie :

$$Q(\theta,\theta^m) = \Big\langle \log f(y,z \mid \theta) \mid y, \theta^m \Big\rangle. \tag{3.11}$$

L'expression 3.11 s'interprète comme l'espérance de la log-vraisemblance complète, espérance prise sur la distribution a posteriori des données manquantes z connaissant les observations x et basée sur les anciens paramètres θ^m.

Chaque itération peut se décomposer en deux étapes [31] :

- Etape **E** : calculer les composantes de l'espérance $Q(\theta, \theta^m)$ qui ne dépendent pas de θ.
- Etape **M** : mettre à jour les paramètres par : $\theta^{m+1} = \arg\max_\theta Q(\theta, \theta^m)$

La croissance de la vraisemblance à chaque itération de cet algorithme connu sous le nom EM (*Expectation-Maximization*) est demontrée dans l'annexe A.

3.6.5 EM et ses variantes appliqués au mélange gaussien

L'algorithme EM est particulièrement adapté pour estimer les paramètres d'un mélange fini de distributions [98] [21]. Dans cette situation, les informations manquantes z sont les classes des observations et les paramètres du modèle sont θ. A l'itération $m+1$, l'espérance à maximiser s'écrit dans ce cas

$$Q(\theta, \theta^m) = \sum_{i=1}^{n} \sum_{j=1}^{J} \underbrace{\left\langle z_{ij} \mid x, \theta^m \right\rangle}_{t_{ij}^m} \log(p_j f(x_i \mid \alpha_j))$$

A l'étape E, on calcul les parties de $Q(\theta, \theta^m)$ qui ne dépendent pas de θ, soit ici, pour $1 \leq i \leq n$ et $1 \leq j \leq J$

$$t_{ij}(\theta^m) = \frac{p_j^m \varphi(x_i \mid \mu_j^m, \Sigma_j^m)}{\sum_{\ell=1}^{J} p_\ell^m \varphi(x_i \mid \mu_\ell^m, \Sigma_\ell^m)}. \tag{3.12}$$

c'est à dire les probabilités d'appartenance a posteriori des observations aux classes connaissant les données x et en se basant sur les paramètres θ^m de l'itération précédente.

L'étape M consiste à chercher

$$\theta^{m+1} = \arg\max_\theta \sum_{\ell=1}^{J} t_{ij}(\theta^m) \log p_j + \sum_{\ell=1}^{J} t_{ij}(\theta^m) \log f(x_i \mid \varphi_j)$$

c'est-à-dire

$$p_j^{m+1} = \arg\max_{p_j} \sum_{j=1}^{J} n_j^{m+1} \log p_j \tag{3.13}$$

$$\forall \;\; 1 \leq j \leq J, \;\; \varphi_j^{m+1} = \arg\max_{\varphi_j} \sum_{i=1}^{n} t_{ij}(\theta^m) \log f(x_i \mid \varphi_j) \tag{3.14}$$

où

$$n_j^{m+1} = \sum_{i=1}^{n} t_{ij}^m(\theta^m)$$

peut s'interpréter comme l'espérance du nombre d'observations appartenant à la classe j. La résolution de l'équation 3.13 sous la contrainte $\sum_{j=1}^{J} p_j = 1$ donne, pour $1 \leq j \leq J$,,

$$p_j^{m+1} = \frac{n_j^{m+1}}{n}.$$

Quant à la recherche des paramètres des classes, chacun de ces J sous problèmes peut être vu comme une maximisation de la vraisemblance des paramètres φ_j à l'intérieur de la classe j, en pondérant chaque individu x_i par son degré d'appartenance t_{ij}^m à cette classe. Par exemple, dans le cas du mélange gaussiens le plus général, les formules obtenues pour recalculer les paramètres peuvent être interpétées comme des estimateurs du maximum de vraisemblance de chaque distribution en pondérant les individus par t_{ij}^m :

$$\mu_j^{m+1} = \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} t_{ij}^m(\theta^m) y_i \tag{3.15}$$

$$\Sigma_j^{m+1} = \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} t_{ij}^{m+1}(\theta^m)(y_i - \mu_j^{m+1})(y_i - \mu_j^{m+1})^{\mathsf{T}} \tag{3.16}$$

La structure d'un lancer de EM est récapitulée dans la figure 3.6.5 pour le cas du modèle de mélange gaussien le plus général.

Convergence. La convergence de EM (voir annexe A) a été établie dans le cadre des modèles de mélange par Redner et Walker [98] sous des conditions asymptotiques assez peu contraignantes. Leur théorème met en relief la nécessité de connaître le nombre de classes et l'importance d'une initialisation qui ne soit pas trop éloignée des vrais paramètres. Redner et Walker soulignent de plus que la convergence de EM, de nature linéaire, est d'autant plus rapide que les composants sont bien séparés. Souvent la solution produite par EM dépend fortement de la position initiale, et l'algorithme EM peut converger vers un palier (non maximum) de la vraisemblance ou rester très longtemps sur un tel palier. De plus, même lorsque EM converge vers un maximum de la vraisemblance, celui-ci n'est que local. Or, dans le cadre des modèles de mélange, la surface de la vraisemblance présente souvent plusieurs maximum locaux [83].

Initialisation. Différentes techniques peuvent être envisagées pour atténuer la dépendance de l'algorithme EM à la position initiale. La tactique la plus fréquement utilisée est similaire à celle décrite pour la méthode de centres-mobiles : on lance plusieurs fois l'algorithme EM de différentes positions initiales θ^0 choisies au hasard, et l'on retient la solution $\hat{\theta}$ qui donne la plus grande vraisemblance $L(\theta)$. Cette tactique est simple à mettre en place, mais elle peut être coûteuse en temps de calcul dans les cas où l'algorithme converge lentement.

Par ailleurs, d'autres techniques itératives d'optimisation peuvent être employées pour tenter de palier la lenteur de l'algorithme. Cependant, nous n'avons pas étudié ce point durant cette thèse. A titre d'exemple, on peut citer la méthode de Newton, qui consiste à mettre à jour le paramètre θ en appliquant la formule suivante :

$m \leftarrow 0$
Initialiser les proportions, centres et variances a une valeurs arbitraire θ^0
répéter
{ Etape **E**: calculer les probabilités d'appartenance t_{ij}^m}
pour i de 1 à n, j de 1 à J faire

$$t_{ij}(\theta^m) \leftarrow \frac{p_j^m \varphi(x_i|\mu_j^m, \Sigma_j^m)}{\sum_{\ell=1}^{J} p_\ell^m \varphi(x_i|\mu_\ell^m, \Sigma_\ell^m)}$$

{ Etape **M**: recalculer les paramètres θ^{m+1}}
pour j de 1 à J faire

$$n_j^{m+1} \leftarrow \sum_{i=1}^{n} t_{ij}^m(\theta^m); \quad p_j^{m+1} \leftarrow \frac{n_j^{m+1}}{n}$$

$$\mu_j^{m+1} \leftarrow \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} t_{ij}^m(\theta^m) y_i$$

$$\Sigma_j^{m+1} \leftarrow \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} t_{ij}^{m+1}(\theta^m)(y_i - \mu_j^{m+1})(y_i - \mu_j^{m+1})^{\mathsf{T}}$$

$m \leftarrow m + 1$
jusqu'à $t^m = t^{m-1}$ *ou* [ou $m = mMAX$]
$\hat{\theta} \leftarrow \theta^m$

FIG. 3.6: *Un lancer de l'algorithme EM*

$$\theta^{m+1} = \theta^m - H_L(\theta^m)^{-1} \nabla_\theta L(\theta^m)$$

où $H_L(\theta^m)^{-1}$ désigne la matrice hessienne de $L(\theta)$ en θ^m. Cette procédure a une vitesse de convergence quadratique, mais elle requiert des calculs importants pour inverser la matrice Hessienne.

Plusieurs tests de convergence de l'algorithme EM peuvent être envisagés. Une stratégie simple consiste à tester la stabilisation des probabilités d'appartenances. Une façon de procéder est de vérifier si la plus grande différence entre les probabilités d'appartenance est inférieure à un certain seuil :

$$\max_{i,j} \mid t_{ij}^{m+1} - t_{ij}^m \mid \; < \epsilon$$

avec par exemple $\epsilon = \frac{0.01}{J}$. Outre sa simplicité de mise en oeuvre, ce test donne lieu à une interprétation intuitive du seuil ϵ. Une stratégie alternative est de tester la stabilisation du critère de vraisemblance, par une condition du type

$$\left| \frac{L(\theta^{m+1}) - L(\theta^m)}{L(\theta^m)} \right| \; < \epsilon$$

avec un seuil ϵ arbitrairement choisi, par exemple $\epsilon = 10^{-6}$. De plus, une condition d'arrêt supplémentaire $m = mMax$ est souvent utilisée afin de ne pas laisser l'algorithme EM séjourner pendant de nombreuses itérations dans un col de la vraisemblance. Ceci peut être utile si l'on envisage de redémarrer l'algorithme un certain nombre de fois et de garder le meilleur résultat. Notons que dans notre implémentation de l'algorithme EM, nous couplons ces deux derniers critères pour tester la convergence. Un dernier test de convergence consiste à tester la stabilisation des paramètres en évaluant la distance entre ces paramètres de deux itérations successives. Cependant ce calcul est relativement complexe à mettre en oeuvre.

Variante CEM. Dans ce travail nous utilisons une variante de EM, connue sous le nom de CEM (classifiante EM) qui converge beaucoup plus rapidement (d'un facteur de 10 au moins) que EM ([21]). L'algorithme CEM cherche à maximiser le critère de vraisemblance classifiante donné par :

$$L_c(z, \theta) = \sum_{i=1}^{n} \sum_{j=1}^{J} z_{ij} \log \{p_j \varphi(y_i|\mu_j, \Sigma_j)\} . \tag{3.17}$$

Il possède une étape intermédiaire C entre les deux étapes de l'algorithme classique EM (algorithme 3.6.5). Cette étape consite à calculer, à partir des probabilités d'appartenance t_{ij}, une classification z par la règle de maximum a posteriori. La figure 3.6.5 résume un lancer de cet algorithme dans le cas d'un modèle gaussien le plus général.

3.6.6 Structures en compétition

L'algorithme EM et ses variantes appliquées sur les mélanges exigent en général que l'utilisateur fixe au préalable le nombre de classes J et les contraintes à imposer sur les paramètres. Or, face à un problème réel et plus particulièrement à notre usage des mélanges gaussiens pour modéliser l'apparence intra-plan d'un objet suivi, on n'a aucune information a priori sur le degré de variabilité de cet objet c'est-à-dire le nombre de classes d'apparences intra-plan de l'objet. En plus, le degré de variabilité d'un objet peut différer d'un objet à un autre. De ce fait, et aussi pour les raisons discutées dans la section 3.6.2.1 il parait très utile d'utiliser des techniques qui permettent de déterminer automatiquement la structure du mélange gaussien c'est-à-dire le meilleur nombre de composantes gaussiennes et le modèle gaussien le plus approprié aux données modélisées.

De nombreuses approches ont été développées pour sélectionner automatiquement la structure de mélange la plus adéquate pour un échantillon donnée: tests d'hypothèse, facteur de Bayes, critères d'information et le critère de vraisemblance classifiante [71] [28] [23] [10]. Le lecteur intéressé pourra se référer à la thèse de Biernacki [9], qui consacre à ce sujet une étude expérimentale approfondie sur des données simulées. Nous décrivons plus particulièrement ici les critères d'information et le critère de vraisemblance classifiante dont nous nous sommes servis dans ce travail pour leur fondement théorique justifiable et leur caractère de simplicité de mise en oeuvre.

$m \leftarrow 0$
Initialiser les proportions, centres et variances a une valeurs arbitraire θ^0
répéter
{ Etape **E**: calculer les probabilités d'appartenance t_{ij}^m}
pour i de 1 à n, j de 1 à J faire

$$t_{ij}(\theta^m) \leftarrow \frac{p_j^m \varphi(x_i|\mu_j^m, \Sigma_j^m)}{\sum_{\ell=1}^{J} p_\ell^m \varphi(x_i|\mu_\ell^m, \Sigma_\ell^m)}$$

{ Etape **C**: calculer la classification z^mpar MAP }
pour i de 1 à n, j de 1 à J faire
$z_i^m \leftarrow e_j$avec
$j = \arg\max_j t_{ij}^m$
{ Etape **M**: recalculer les paramètres θ^{m+1}}
pour j de 1 à J faire

$$n_j^{m+1} \leftarrow \sum_{i=1}^{n} z_{ij}^m(\theta^m);$$

$$p_j^{m+1} \leftarrow \frac{n_j^{m+1}}{n}$$

$$\mu_j^{m+1} \leftarrow \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} z_{ij}^m(\theta^m) y_i$$

$$\Sigma_j^{m+1} \leftarrow \frac{1}{n_j^{m+1}} \sum_{i=1}^{n} z_{ij}^{m+1}(\theta^m)(y_i - \mu_j^{m+1})(y_i - \mu_j^{m+1})^{\mathsf{T}}$$

$m \leftarrow m + 1$
jusqu'à $z^m = z^{m-1}$ *ou* [ou $m = mMAX$]
$\hat{z} \leftarrow z^{(m-1)}$
$\hat{\theta} \leftarrow \theta^m$

FIG. 3.7: *Un lancer de l'algorithme CEM.*

On peut voir que ces critères comme une forme de vraisemblance pénalisée, ceci dans le cadre d'une estimation des paramètres par maximum de vraisemblance [28]. Pour comprendre leur principe, on peut tout d'abord remarquer que dans un tel cadre, la vraisemblance $L(\hat{\theta}_M)$ des paramètres estimés tend à croître avec le nombre de degrés de libertés du modèles M. En effet, soit un modèle M_A dont le domaine des paramètres libres $\Omega(M_A)$ est inclus dans celui $\Omega(M_B)$ d'un autre modèle plus général M_B. Alors le vecteur de paramètre $\hat{\theta}_{M_A}$ qui maximise la vraisemblance dans le domaine $\Omega(M_A)$ appartient au domaine $\Omega(M_B)$. Sa vraisemblance $L(\hat{\theta}_{M_A})$ est donc inférieure ou égale à celle du vecteur des paramètres $L(\hat{\theta}_{M_B})$ qui maximise la vraisemblance sur le domaine $\Omega(M_B)$. En notant

$$L_{\max}(M) = L(\hat{\theta}_M) \text{ où } \hat{\theta}_M = \arg \max_{\theta \in \Omega(M)} L(\theta)$$

on a donc

$$\Omega(M_A) \subset \Omega(M_B) \Rightarrow L_{\max}(M_A) \leq L_{\max}(M_B).$$

De ce qui précède, on peut déduire que la vraisemblance croît avec le nombre de classes lorsqu'on considère des modèles de mélange à proportions variables. Ainsi, bien que la vraisemblance donne une mesure de l'ajustement du modèle aux données, elle ne permet pas de sélectionner directement le modèle le plus adéquat pour un échantillon donné.

Le principe de des critères d'information consiste à choisir le modèle qui fait croître la vraisemblance le plus possible, tout en minimisant la complexité du modèle. Pour cela, la plupart des critères se basent sur le maximum de vraisemblance pénalisé par le nombre de paramètres libres du modèle, ce qui donne l'expression généralement suivante à maximiser sur les différents modèles en compétition :

$$CI(M) = -2L_{max}(M) + \tau_{CI}Q(M)$$

La fonction $Q(M)$ indique le nombre de paramètres libres du modèle M, c'est-à-dire la dimension du domaine $\Omega(M)$. Elle représente la complexité du modèle M. Le coefficient τ_{CI} représente la pénalisation de la complexité du modèle spécifique au critère CI. Par exemple, le critère *Akaike information criterion* (AIC) proposé par Akaike [2] s'écrit :

$$AIC(M) = -2L_{max}(M) + 2Q(M) \tag{3.18}$$

on a donc $\tau_{CI} = 2$. Bozdogan [15] propose une variante du critère d'Akaike appelée AIC3:

$$AIC(M) = -2L_{max}(M) + 3Q(M) \tag{3.19}$$

donc $\tau_{CI} = 3$. Le critère *Bayes information criterion* (BIC) est obtenu par Schwarz [111] comme une approximation de la solution bayésienne exacte au problème de sélection de modèle :

$$BIC(M) = -2L_{max}(M) + \log(n)Q(M) \tag{3.20}$$

Ici, la pénalisation $\tau_{CI} = \log(n)$ fait intervenir le nombre d'observations.

Biernacki et al. dans [10] proposent d'utiliser la vraisemblance classifiante pour pénaliser le modèle gaussien et le nombre de composantes gaussiennes. Ce critère dénoté par ICL est donné par l'équation 3.21. Sur des données simulées dans la dimension deux Biernacki et al. prouvent que ICL est mieux adapté que d'autres critères lorsque la forme de ces données n'est pas gaussienne.

$$ICL(M) = -2L_M + Q_M ln(n) - 2\sum_{i=1}^{n}\sum_{j=1}^{J}\hat{c}_{ij}t_{ij}, \tag{3.21}$$

où $\hat{c}_{ij}$ représente la partition estimée déduite des probabilités a posteriori t_{ij}.

A la différence des critères d'informations cités jusqu'ici, Celeux et al. [23] proposent d'utiliser l'entropie $E(M) = -\sum_{i=1}^{n}\sum_{j=1}^{J} t_{ij}\log t_{ij} \geq 0$ dans la sélection de la structure du

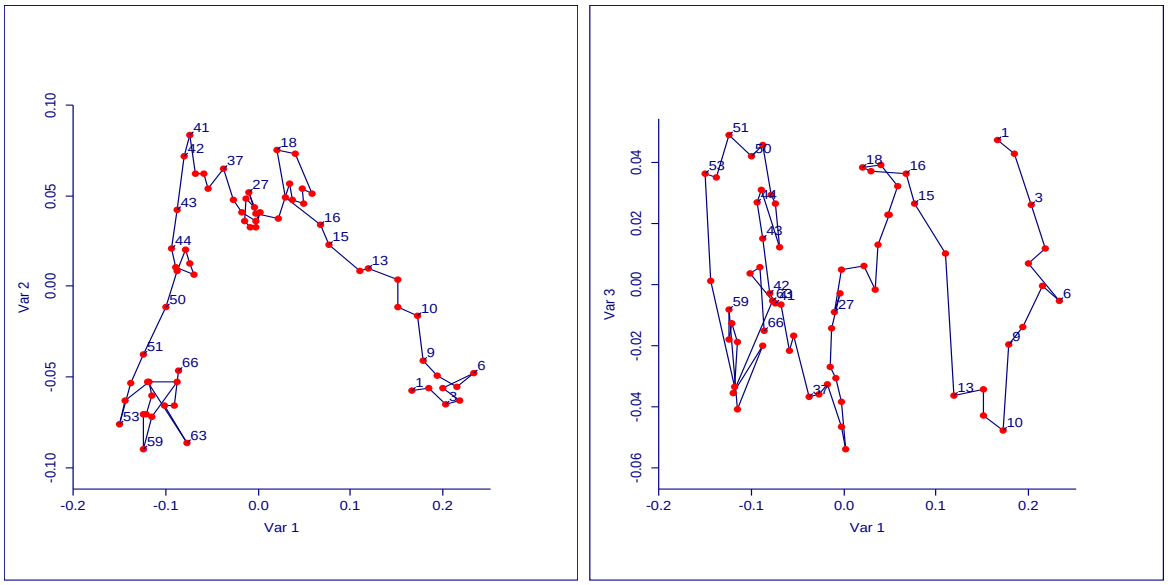
mélange gaussien. L'entropie est vue comme une mesure de la capacité de fournir une partition adéquate des données par un mélange gaussien de J-composantes. Ce critère à minimiser est donné par :

$$NEC(M) = \frac{E(M)}{L_M - L_M^1} \tag{3.22}$$

où L_M^1 dénote la vraisemblance maximale pour une seule gaussienne.

Comme nous le verrons dans le chapitre suivant (partie expérimentale et discussion), le choix d'un critère influence considérablement les résultats de classements des objets : mieux la variabilité intra-plan est modélisée, plus l'appariement et le classement des objets sont bons. Des taux de bon classements du même ordre de grandeur sont obtenus lorsque ICL ou BIC sont employés. Par contre le critère NEC fournit des mauvais résultats surtout lorsque la variabilité d'un objet suivi est modélisée par une seule gaussienne. Ce qui peut être expliqué par la normalisation de l'entropie par le terme L_M^1.

FIG. 3.8: *Sous ensemble (12/66) des apparences intra-plan de l'objet suivi "voiture Ford" de la séquence vidéo "Avengers".*



(a) (b)

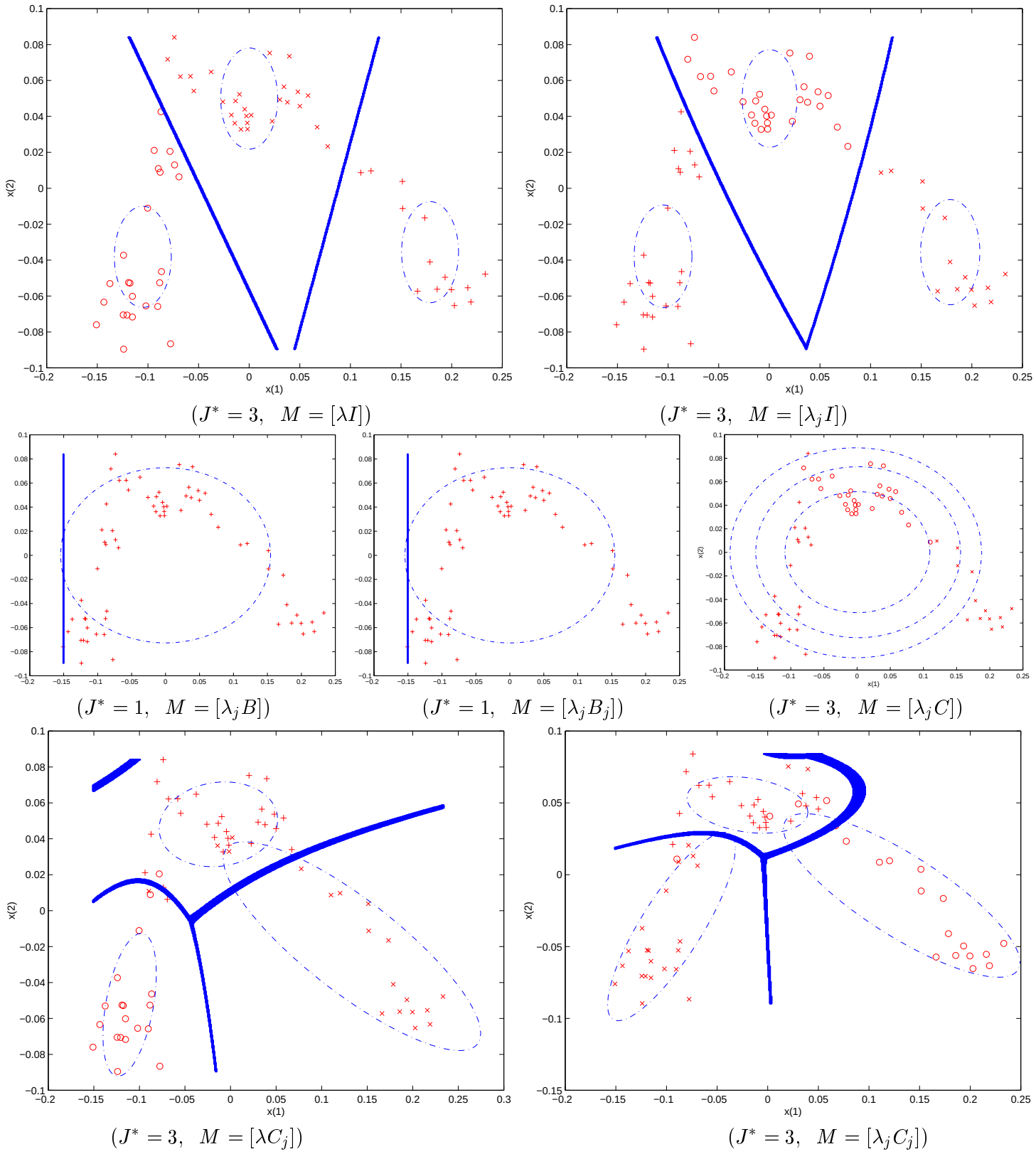
FIG. 3.9: *Représentation des histogrammes de couleurs de l'objet suivi "voiture Ford" (figure 3.8) dans le premier (a) et le deuxième plan factoriel (b) de l'espace réduit à* 10 *dimensions. Un segment de droite relie deux apparences successives.*

3.7 Expérimentation

Cette section décrit une étude expérimentale de l'approche de modélisation statistique de la variabilité intra-plan des objets suivis. Dans cette démarche on procède de la manière suivante. Un objet suivi de n apparences dans le plan vidéo, est décrit par un nuage de n point dans l'espace de descripteurs "D" de dimension d. Dans l'espace de descripteurs réduit par ACP de dimension d_E selon une qualité de représentation dans cet espace de 95%, on modélise la variabilité intra-plan. Dans une **première expérience** on fixe le modèle gaussien M_i ($i = 1..7$) et on fait varier seulement le nombre de composantes gaussiennes de 1 à une borne supérieure (MaxNbC) fixée à priori. Dans une **seconde expérience** on fixe le nombre de composantes gaussiennes de mélange gaussien et on fait varier le modèle M_i dans la famille des modèles. Dans une **troisième expérience** on fait varier ces deux éléments du mélange gaussien. On utilise dans ces expériences le critère ICL qui, d'après la discussion ci-dessus, est le critère théorique le mieux adapté pour le choix de la structure du mélange.

Pour valider cette démarche, on présente ici seulement les résultats sur l'objet suivi "voiture Ford" de 66 apparences, extrait du corpus vidéo "Avengers". Des expérimentations intensives ont été réalisées dans les deux chapitres suivants pour un objectif d'appariement et de classification d'objets. La figure 3.8 illustre quelques unes des apparences intra-plan de la "voiture Ford". On extrait de ces apparences des histogrammes de couleurs, calculés dans l'espace RGB, de 64 cellules

La figure 3.9 montre la distribution de cet objet suivi dans les deux premiers plans

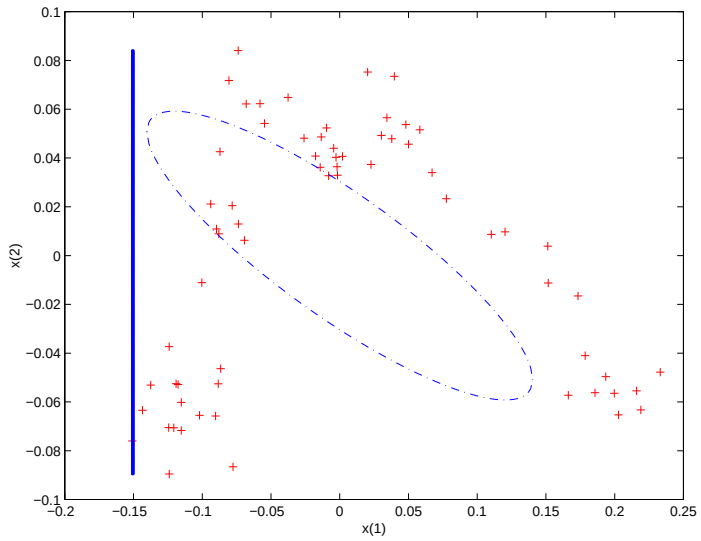


$(J^* = 3, \quad M = [\lambda I])$ $(J^* = 3, \quad M = [\lambda_j I])$

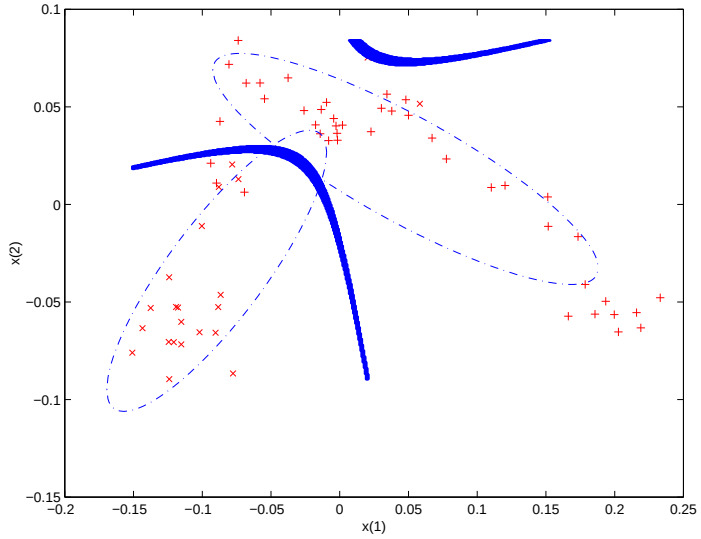
$(J^* = 1, \quad M = [\lambda_j B])$ $(J^* = 1, \quad M = [\lambda_j B_j])$ $(J^* = 3, \quad M = [\lambda_j C])$

$(J^* = 3, \quad M = [\lambda C_j])$ $(J^* = 3, \quad M = [\lambda_j C_j])$

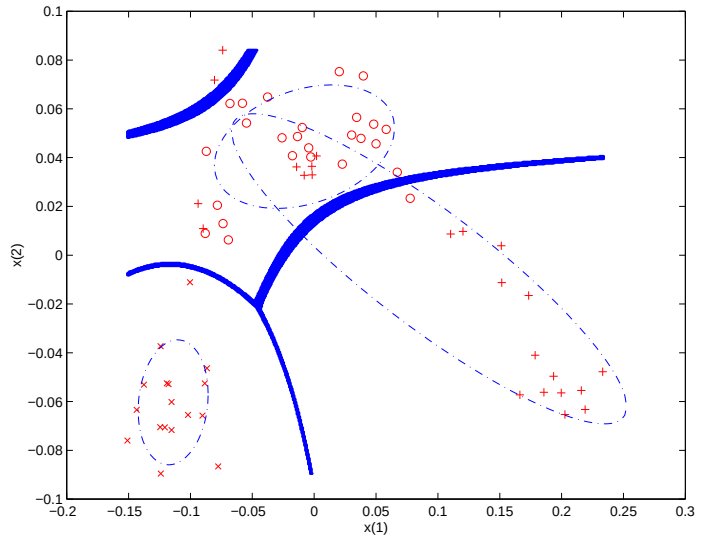
FIG. 3.10: *Illustration des résultats de modélisation des données de la figure 3.9.a, dans le cas où le modèle gaussien M est connu a priori et où seul le nombre de composantes gaussiennes est en compétition. Le nombre choisi par ICL est désigné par J^*. Pour chaque modèle la partition ainsi que les ellipses de variances et les frontières sont illustrées.*



$(J = 1, \quad M^* = [\lambda C_j])$

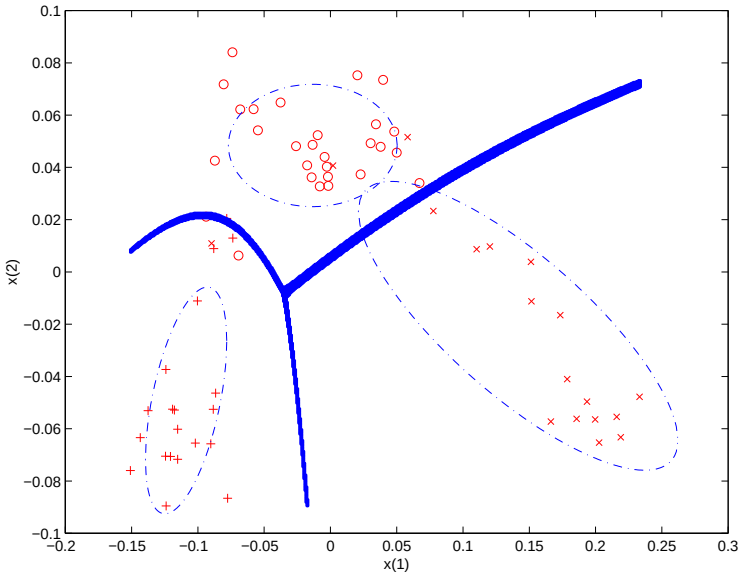


$(J = 2, \quad M^* = [\lambda C_j])$



$(J = 3, \quad M^* = [\lambda_j C_j])$

FIG. 3.11: *Illustration des résultats de modélisation des données de la figure 3.9.a dans le cas où le nombre de composantes gaussiennes J est connu a priori et où seul les modèles gaussiens sont mises en compétition. Le modèle sélectionné par ICL est désigné par M^*.*



$(J^* = 3, \quad M^* = [\lambda C_j])$

FIG. 3.12: *Illustration des résultats de modélisation des données de la figure 3.9.a dans le cas où les modèles gaussiens et le nombre de composantes gaussiennes sont mises en compétition. La structure sélectionnée par ICL est désignée par* $(J^*, \ M^*)$.

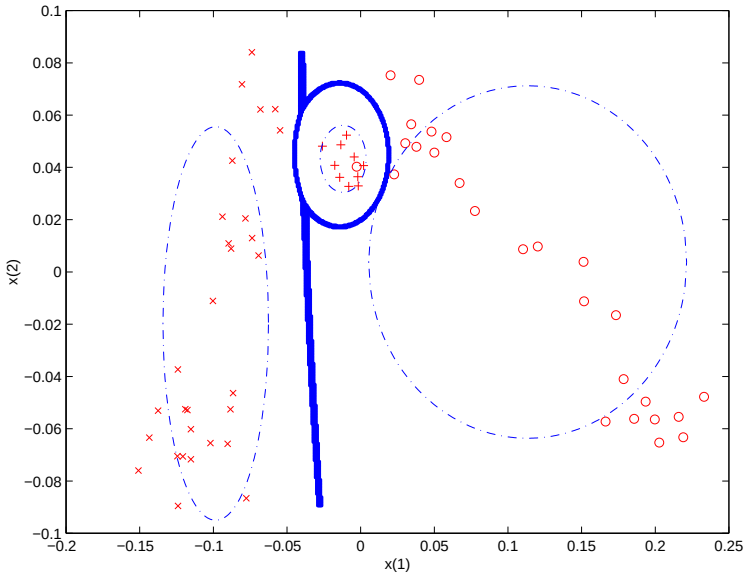
factoriels de l'espace réduit de dimension $d_E = 10$. Lors des trois expériences décrites précédemment on fixe MaxNbC à 3 pour l'exemple de cette figure. Ce nombre présente le vrai nombre de classes d'apparences intra-plan (voir figure 3.8) qu'on souhaite retrouver automatiquement par le critère ICL. D'autre part, il reste raisonnable vu le petit nombre d'apparences de l'objet modélisé.

Les figures 3.10, 3.11 et 3.12 illustrent les résultats de la modélisation lors de la première, deuxième et troisième expérience respectivement. Les partitions obtenues ainsi que les ellipses de variances et les frontières inter-classes d'apparences sont projetées dans le premier plan factoriel seulement. Rappelons que la modélisation est effectuée dans un espace de descripteurs à 10 dimensions.

Commentaires La distribution multimodale de la "voiture Ford" dans le premier plan factoriel est bien décomposable en trois classes gaussiennes.

Les modèles gaussiens avec contraintes ont été introduits dans ce travail pour palier le problème de la surestimation des paramètres gaussien du modèle général lorsque l'objet suivi dispose de peu d'apparences intra-plan par rapport à la dimension de l'espace de descripteurs. D'un autre coté, le mélange gaussien dispose d'un ensemble de critères qui permet de sélectionner la structure la plus adaptée aux données.

Les résultats de la première expérience (figure 3.11) montrent que les deux modèles gaussiens diagonaux $\lambda_j B$ et $\lambda_j B_j$ sont mal adaptés aux données. Ces modèles imposent une orientation vers les axes (figure 3.4). Ces contraintes sont difficiles à réaliser sur les



$(J^* = 3, \quad M^* = [\lambda_j B_j])$

FIG. 3.13: *Une partition alternative de celle illustrée dans la figure 3.12, obtenue en partant d'une initialisation de EM très différente.*

données modélisées. Il en résulte une mauvaise sélection du nombre de classes par le critère ICL. Pour la même raison, on trouve que le modèle $\lambda_j C$ est mal adapté à ces données car il imposent d'une même orientation et des formes ellipsoïdes égales pour les trois classes choisies par ICL. Les modèles restant fournissent des partitions proches des vraies partitions, avec des frontières linéaires et quadratiques. La frontière optimale est fournie par le modèle le plus général.

Les résultats de la deuxième expérience (figure 3.10) montrent que lorsque le nombre de classes est fixé à 2 et 3 les deux modèles $\lambda_j C$ et $\lambda_j C_j$ sont sélectionnés respectivement. Dans le cas où le nombre de classes est fixé à 2, on peut nettement voir que le modèle général est bien adapté aux données.

Dans la troisième expérience (figure 3.12) la structure du mélange gaussien semble être bien adaptée aux données "voiture Ford". Trois classes d'apparences ont été identifiées et les trois matrices de variances ont des volumes égaux mais des orientations et des formes différentes.

En effet, l'adaptation d'une structure fixe de la loi du mélange sur des données de complexité variable n'est pas la bonne stratégie à appliquer. Par ailleurs le critère comme ICL et BIC peuvent palier cet handicap. Cependant ces critères ne garantissent pas toujours la sélection de la meilleure structure de la loi de mélange. Par exemple, la figure 3.13 décrit une deuxième partition obtenue par ICL lorsque le nombre d'initialisation de l'algorithme EM est changé, ce qui conduit à une convergence assez différente. La convergence de EM et la dépendance de la solution initiale ont été déjà discutées dans la section 3.6.5.

3.8 Conclusion

Dans ce chapitre, nous avons abordé en détail le problème de la variabilité intra-plan d'un objet suivi dans l'espace de descripteurs. Cette variabilité intra-plan est dûe principalement aux changements d'apparence de l'objet à travers le temps et au manque de robustesse des descripteurs classiques à ces changements de l'image. Ces deux facteurs conduisent à une représentation spatiale dans l'espace de descripteurs qui n'est pas compacte et souvent multimodale. Une telle représentation représente un handicap majeur pour tout processus de comparaison de deux objets suivis – apparier toutes les apparences deux à deux et comment prendre une décision ? ou bien apparier juste les deux apparences représentatives des objets suivis ? –. Pour permettre une comparaison fiable de deux objets nous avons proposé ici de représenter un objet suivi par des modèles statistiques. D'une part ces modèles capturent la variabilité intra-plan et d'autre part ils ont une taille raisonnable. Il s'agit ici des modèles gaussiens multivariés qui représentent les classes d'apparences intra-plan des objets suivis. Nous avons adopté le mélange gaussien pour ses propriétés intéressantes vis-à-vis du problème étudié :

- il est suffisamment général pour modéliser des distributions complexes;
- il est suffisamment efficace pour modéliser une distribution avec peu ou beaucoup d'observations;
- il possède des critères qui permettent de sélectionner la structure du mélange gaussien (modèle gaussien et nombre de composantes gaussiennes) automatiquement.

Ce chapitre a décrit notre sélection des descripteurs de bas niveaux pour caractériser les apparences d'un objet suivi : les histogrammes de couleurs bruts et normalisés. Les expérimentations illustrées ici nous permettent de tirer les conclusions suivantes :

- la distribution d'un objet suivi dans l'espace des histogrammes de couleurs est multimodale et correspond bien à un mélange de lois gaussiennes;
- les modèles gaussiens qui imposent des contraintes sur l'orientation de la matrice de variance sont moins adaptés en pratique que les autres;
- les critères du mélange gaussien permettent un choix d'une structure de mélange gaussien qui est bien adaptée aux données modélisées mais qui n'est pas toujours optimale.

L'extension d'une telle approche sur des descripteurs locaux autour des points d'intérêts extraits des apparences est une tâche délicate car il s'agit dans un premier temps de suivre un point d'intérêt d'une apparence à une autre sachant que l'objet est non rigide; dans un second temps il faut construire une loi de mélange pour chaque point suivi, et enfin mettre en correspondance un point requête avec les n lois de mélanges des n points d'intérêts de l'objet.

Chapitre 4

Classification supervisée des objets vidéo

On va relativiser les "urgences".

Pour créer un film interactif et créer des liens entre des objets "identiques", il faut être capable de repérer les occurrences de ces objets à travers la vidéo. Ce chapitre décrit notre approche de classification semi-automatique des objets suivis, où les classes d'objets d'un film vidéo donné sont connues a priori. Nous appliquons la théorie de mélange gaussien discutée dans le chapitre précédent dans le cas de la recherche automatique des classes d'apparences intra-plan. Cette approche s'applique aussi à la classification discriminante de tous les objets du film dans les classes d'objets pré-définis.

4.1 Introduction

L'étude de la variabilité des objets suivis du chapitre précédent à été menée principalement pour donner des éléments de réponse satisfaisants à la question suivante : comment apparier deux objets suivis ? Les objets suivis étant non-rigides et en mouvements à travers le temps, ils subissent plusieurs changements d'images significatifs, ce qui provoque leur variabilité intra-plan dans l'espace de descripteurs. La généralisation de la méthode des "images-clés" (une image représentative par plan) pour apparier deux objets suivis sur la base de leurs deux occurrences représentatives (occurrence médiane par exemple) parait inacceptable vu la variabilité intra-plan des objets suivis et la difficulté même d'identifier correctement les occurrences d'un même objet suivi (voir l'exemple de la figure 3.2). Capturer la variabilité intra-plan par un mélange gaussien est une meilleure solution pour représenter l'objet suivi; des classes d'apparences intra-plan sont d'abord identifiées et ensuite représentées dans un nouvel espace : l'espace des paramètres gaussiens. En effet, une façon pour apparier deux objets suivis est de prédire les probabilités que n observations provenant de la loi de mélange du deuxième objet suivi soient des réalisations de la loi

de mélange du premier objet suivi, ou bien de calculer directement une distance adaptée entre les classes d'apparences intra-plan de ces objets. Il faudra bien sûr accepter que la méthode échoue si les apparences d'un objet dans deux plans ne sont pas similaires. Par exemple, lorsque le même acteur est habillé différemment dans deux endroits du film.

Ainsi, une stratégie de classification des objets suivis se compose d'abord d'une phase de mise en correspondance des apparences de ces objets, et ensuite d'une autre phase de regroupement des objets similaires dans des classes d'objets.

Dans ce chapitre nous proposons une approche de classification supervisée des objets vidéo où on considère les classes d'objets d'un film donné a priori connues. Ici, le constructeur de la vidéo interactive, nommé dans la suite auteur de la vidéo hyperliée, désigne quelques objets différents dans le film vidéo comme "modèles d'objets suivis". Ainsi, chaque modèle d'objet suivi est modélisé d'abord par une loi de mélange gaussien dont le nombre des composantes et le modèle gaussien sont sélectionnés automatiquement en fonction de son degré de variabilité (voir chapitre précédent). Ensuite, la loi globale du mélange de tous les modèles d'objets suivis est construite pour pouvoir classer tout occurrence d'un objet dans le film vidéo, dans la classe d'objets la plus probable a posteriori – règle du maximum a posteriori (classement individuel des apparences). L'étape finale consiste à classer un objet suivi requête dans l'une des classes de modèles d'objets selon le vote majoritaire de toutes ces occurrences classées séparément (classement robuste des apparences).

L'organisation de ce chapitre est le suivant. La section suivante présente un état de l'art des différents types d'applications utilisant une approche de classification supervisée basée sur le mélange gaussien. Les différentes étapes de l'approche proposée sont détaillées dans la section 4.3. Cette approche est validée par des expérimentations réelles sur la séquence vidéo "Avengers-1" qui comporte environ 1391 objets segmentés. Cette séquence est extraite du corpus vidéo de l'INA (voir section 2.5). Une variante de descripteurs globaux de la couleurs est utilisée pour décrire les apparences intra-plan de ces objets. L'évaluation globale des résultats expérimentaux ainsi que l'étude comparative avec l'approche d'indexation classique de "moyenne temporelle de descripteurs" [133] sont détaillées dans la section 4.4. Cette étude montre une augmentation significative de 10% à 35% des pourcentages de bon classement des requêtes par l'approche proposée relativement à l'approche classique. Nous concluons ce chapitre dans la section 4.7.

4.2 État de l'art

Quand il s'agit d'identifier ou de classifier des scènes de visages, plusieurs aspects d'apparences doivent être pris en compte lors de l'étape d'apprentissage du système : visage avec ou sans lunettes, barbe, moustache, face gauche/droite, etc. Sung et Poggio [119] considèrent dans leur système six classes de visages différentes, chacune étant modélisée par une gaussienne. Ils utilisent l'algorithme de centres mobiles pour fabriquer ces classes et la distance de Mahalanobis pour classer les nouveaux visages. Dans un deuxième temps, ils introduisent une classe de rejet (bâtiments, ...) pour raffiner les résultats de détection de visages.

Pour le même objectif, mais cette fois ci sur les séquences d'images, McKenna et al. [82] utilisent le mélange gaussien pour suivre et classifier les visages. Les visages détectés dans les plans sont caractérisés par des distributions de couleurs et parmi ces visages suivis quelques uns sont pris comme modèle et serviront à l'apprentissage du système. Cette approche est la plus proche de celle proposée dans ce chapitre; les visages suivis sont des cas particuliers des objets non rigides suivis dans les plans vidéo; les variations d'un objet quelconque sont beaucoup plus complexes que pour celles associées aux visages. Aussi, plusieurs aspects techniques distinguent notre approche de celle de McKenna et al. et en particulier le choix de la structure de la loi de mélange et les données. Dans [82], les expérimentations ont été menées sur une séquence vidéo fabriquée par les auteurs et sont composées d'une quinzaine de plans. Un taux de bon classement des visages autour de 90% a été obtenu sur cette séquence.

L'utilisation du mélange gaussien commence à s'étendre dans la communauté de vision. Pour le problème de la reconnaissance de gestes, Rosales [104] étudie la performance de plusieurs approches de classification supervisée: K-plus proches voisins, gaussien et mélange gaussien. Chaque action humaine parmi les 7 classes différentes de la base d'apprentissage est caractérisée par des descripteurs simples de mouvements nommés par *Motion History Images* et *Motion Energy Images*, un descripteur étant basé sur la différence des intensités de pixels. Ces expérimentations montrent que le mélange gaussien donne les meilleurs résultats par rapport aux autres méthodes. L'approche de Rosales est en réalité une extension des travaux de Davis [29], où ce dernier utilise les k-plus proches voisins au lieu du mélange gaussien.

4.3 Notre approche

4.3.1 Sélection des modèles d'objets suivis

Les objets localisés dans une séquence vidéo correspondent en réalité à un nombre assez petit de classes d'objets homogènes et distinctes les unes des autres. L'approche que nous proposons dans ce chapitre pour classifier les objets vidéo est semi-automatique. Ainsi, une connaissance a priori d'un échantillon de chaque classe d'objets à fabriquer est fournie. Ici on entend par échantillon un objet suivi (au moins), et donc toutes ses occurrences dans le plan. A cette étape interactive, l'auteur de la vidéo hyperliée intervient pour désigner un modèle d'objet par classe d'objets. Ceci est établi dans le système développé pour ce travail d'une manière interactive et simple [56] : par un simple clic de la souris sur l'image de l'objet d'intérêt, affichée dans la mosaïque de plans de la vidéo (une image représentative par plan), le système enregistre l'objet suivi correspondant comme un modèle d'objet suivi (le système est décrit dans le chapitre 8). La figure 4.1 illustre un exemple de quatre modèles d'objets suivis sélectionnés dans la mosaïque des plans de la séquence vidéo Avengers.

4.3.2 Partition de l'espace de descripteurs

Soit L l'ensemble des modèles d'objets suivis désignés par l'utilisateur expert pour une séquence vidéo donnée. Chaque modèle d'objet suivi représente une classe d'ob-

FIG. 4.1: *Sélection interactive des modèles d'objets suivis dans la mosaïque de plans de la séquence Avengers; l'image médiane de chaque plan est affichée et les contours des modèles d'objets sélectionnés sont aussi dessinés.*

jets; et chaque occurrence est caractérisée par un vecteur descripteur de dimension d, et donc représentée par un point multidimensionnels, $y_i \in \Re^d$. Soit $y = (y_1, ..., y_n)$ les données (observations) collectées de L classes d'objets et $z = (z_1, ..., z_n)$ leurs labels d'appartenance à ces classes: y_i est classé dans la ℓ^{eme} classe d'objets si et seulement si $z_i = \ell$, $(\ell \in \{1, \ldots, L\})$. Le nombre d'observations qui appartiennent à la ℓ^{eme} classe (nombre d'occurrences du ℓ^{eme} modèle d'objet suivi) est donné par $n_\ell = \sum_{i=1}^{n}(z_i = \ell)$. La figure 4.2.a illustre les données simulées de deux objets différents dans le plan $\Re^2$. Nous supposons dans la suite que ces observations et leurs labels sont des réalisations indépendamment et identiquement distribuées (i.i.d.) de n couples de variables aléatoires $(\mathbf{Y}, \mathbf{Z}) = \big((Y_1, Z_1), ..., (Y_n, Z_n)\big)$.

La distribution du couple de variables aléatoires $(\mathbf{Y}, \mathbf{Z})$ est un mélange de L distributions (L modèles d'objets suivis). Nous supposons que ces L distributions sont caractérisées

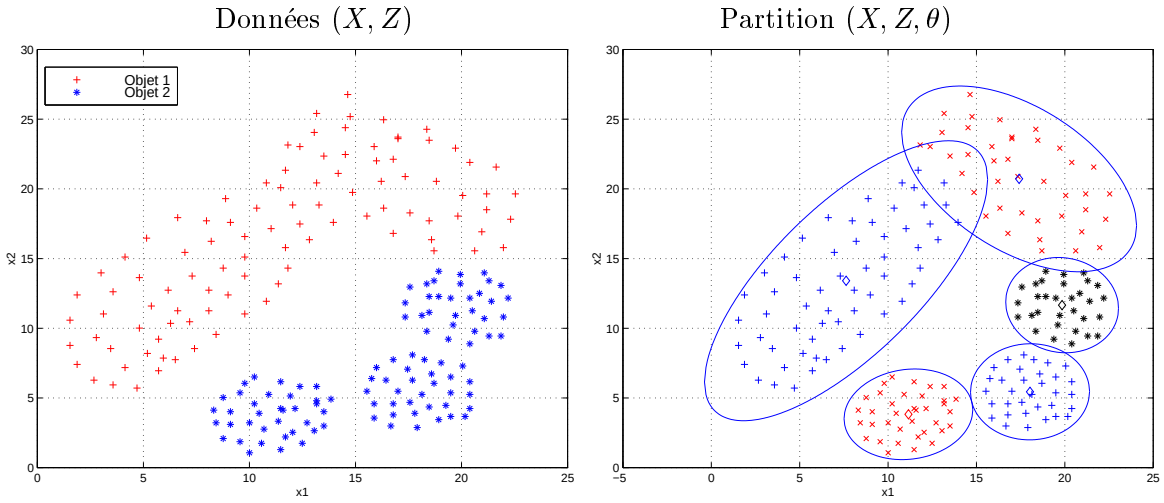


FIG. 4.2: *Illustration de la distribution simulée de deux objets suivis dans l'espace $\mathbb{R}^2$ et de leurs classes d'apparences intra-plan déterminées automatiquement. (a) Les données (X, Z), avec $x_i \in \Re^2$, $z_i = \{1, 2\}$, $L = 2$ et le nombre d'observations de l'objet ℓ est $n_\ell = 100$, $\ell = \{1, 2\}$. (b) La partition, θ_ℓ, qui correspond à chaque objet suivi, déterminée par l'algorithme EM, où, le modèle gaussien général est utilisé, le nombre maximal de composantes gaussiennes (classes d'apparences intra-plan) est fixé à 3, le critère d'information utilisé étant BIC et les nombres de classes retenues étant $J_1 = 2$ et $J_2 = 3$ pour le premier et le deuxième objet suivi respectivement. Les ellipses de variances sont affichées pour les classes d'apparences intra-plan des deux objets.*

par des fonctions de densité de probabilité $\Phi(y \mid \theta_\ell)$, où $1 \leq \ell \leq L$, et $\theta = (\theta_1, ..., \theta_L)$ représentent les paramètres inconnus de ces distributions qui sont mélangées selon des proportions respectives $p_1, ..., p_L$, avec $p_l = \frac{n_\ell}{n}$ (par exemple proportion libre).

La connaissance du paramètre θ produit une partition de l'espace $\mathbb{R}^d$ en L classes, ce qui nous permet alors de **prédire** la classe de toute nouvelle occurrence y d'un objet suivi dans la vidéo (i.e. le vecteur descripteur). Ceci est le principe de la discrimination exclusive où on estime une règle de classement $\hat{u}$ de la forme :

$$\begin{array}{ccc} \hat{u} : \mathbb{R}^d & \longrightarrow & \{1, \ldots, L\} \\ y & \rightarrow & \hat{u}(y) \end{array} \tag{4.1}$$

Nous développons dans la suite le principe d'estimation du paramètre θ_ℓ, $1 \leq \ell \leq L$ à partir de données d'apprentissage $(\mathbf{Y}, \mathbf{Z})$ et nous reviendrons après sur la définition de la règle de classement employée par l'approche proposée.

4.3.3 Estimation des paramètres

L'étude détaillée dans le chapitre précédent sur la variabilité intra-plan des descripteurs montre que la distribution d'un objet suivi est souvent multimodale. Une approximation

par un mélange gaussien multivarié

$$\Phi(y, \theta_\ell) = \sum_{j=1}^{J_\ell} \rho_j \varphi(y \mid \alpha_j)$$

a été proposée pour capturer cette variabilité. Chaque composante gaussienne, $\varphi(. \mid \alpha_j)$, représente une classe d'apparence intra-plan du l^{eme} modèle d'objet suivi. Elle est caractérisée par un centre μ_j, une dispersion Σ_j autour de ce centre, $\alpha_j = (\mu_j, \Sigma_j)$, et par une proportion ρ_j. L'estimation du paramètre du mélange $\theta_\ell = (\rho_1, \ldots, \rho_j, \alpha_1, \ldots, \alpha_{J_\ell})$ est achevée par la méthode de maximum de vraisemblance dans l'algorithme EM (voir section 3.6.4). Le nombre de classes d'apparences intra-plan n'est pas fixé a priori ni le modèle gaussien qui modélise au mieux la distribution. Ils sont déterminés automatiquement par le moyen des critères d'informations comme BIC et ICL (voir section 3.6.6). Les figures 4.2.a et 4.2.b illustrent respectivement les données simulées de deux objets suivis différents et leurs partitions obtenues dans l'espace $\mathbb{R}^2$.

4.3.4 Loi du mélange global

Soient $K = \sum_{\ell=1}^{L} J_\ell$ le nombre total de classes d'apparences intra-plan trouvées pour les L modèles d'objets suivis ($K \geq L$), avec J_ℓ le nombre de composantes gaussiennes retenue automatiquement pour la ℓ^{eme} distribution d'objets suivis, et $\theta = (\theta_1, ...\theta_L)$ leurs paramètres estimés séparément. Seules les proportions sont recalculées en fonction de la nouvelle partition de l'espace $\Re^d$: $p_k = \frac{n_k}{n}$ avec n_k la taille de la k^{eme} classe d'apparence intra-plan et n la taille de données $\mathbf{X}$.

En conséquence le couple de variable aléatoire $(\mathbf{Y}, \mathbf{Z})$ suit une loi du mélange gaussien global de K composantes et de paramètre θ. En un point $y \in \mathbb{R}^d$ cette densité du mélange gaussien global est donnée par :

$$\begin{aligned} f(y|z,\theta) &= \sum_{\ell=1}^{L} \frac{n_\ell}{n} \Phi(y, \theta_\ell) \\ &= \sum_{k=1}^{K} p_k \varphi(y|\mu_k, \Sigma_k) \end{aligned} \tag{4.2}$$

Les seuls paramètres à estimer à nouveau sont les proportions de la loi du mélange global.

4.3.5 Classement individuel des apparences d'objets suivis

De là on déduit les probabilités a posteriori que y_i appartienne aux L classes d'objets. Soient Ω_ℓ la ℓ^{eme} classe d'objets (ℓ^{eme} modèle d'objet suivi avec $\ell \in \{1, \ldots, L\}$) et $A_{\ell_1}, \ldots, A_{\ell_{J_\ell}}$, avec $\ell_j \in \{1, \ldots, K\}$, ses classes d'apparences intra-plan qui constituent une partition de Ω_ℓ modélisée par un mélange gaussien de J_ℓ composantes :

Ayant estimé le paramètre θ du mélange global, nous pouvons calculer pour chaque nouvelle occurrence y_i de $\mathbb{R}^d$ ($1 \leq i \leq n_r$, avec n_r le nombre total des occurrences de l'objet

suivi requête) une probabilité a posteriori qu'elle appartienne à la classe d'apparence intra-plan k ($k \in \{1, \ldots, K\}$) par la formule usuelle de Bayes :

$$t_{ik}(\theta) = Prob(Z = k \mid y_i, \theta) = \frac{p_k \varphi(y_i \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} p_j \varphi(y_i \mid \mu_j, \Sigma_j)}. \tag{4.3}$$

$$\begin{cases} \forall i \neq j \quad A_{\ell_i} \cap A_{\ell_j} = \oslash \\ \cup A_{\ell_j} = \Omega_\ell \end{cases} \tag{4.4}$$

les parties A_{ℓ_j} sont disjointes, alors

$$Prob(\Omega_\ell) = \sum_{j=1}^{J_\ell} Prob(A_{\ell_j})$$

et la probabilité a posteriori que y_i appartienne à la classe d'objets ℓ est

$$t_{i\ell}(\theta) = \sum_{j=1}^{J_\ell} Prob(Z = \ell_j \mid y_i, \theta). \tag{4.5}$$

Après le calcul de tous les $t_{i\ell}(\theta)$, $\ell = 1, \ldots, L$, de la même manière que précédemment, on affecte l'occurrence y_i à la classe la plus probable a posteriori : c'est la règle du *maximum a posteriori* connue par MAP (voir figure 4.3). On note dans la suite par $ci.\%$ le pourcentage de bon classement individuel des apparences d'objets.

Justification du MAP. La théorie Bayésienne de décision justifie la discriminance du MAP en terme de classement. Soit une règle de classement u et soit la fonction $\Im$ de coût 0-1 associant un coût nul aux bons classements et 1 aux mauvais :

$$\begin{cases} \Im(i \mid i) = 0 & i = 1, \ldots, K \\ \Im(i \mid j) = 1 & i = 1, \ldots, K, \quad i \neq j \end{cases} \tag{4.6}$$

Le risque conditionnel s'écrit

$$\begin{aligned} \mathcal{R}(u \mid y) &= \sum_{k=1}^{K} \Im(u(y) \mid k) t_k(y \mid \theta) \\ &= \sum_{k=1, k \neq u(y)}^{K} t_k(y \mid \theta) \\ &= 1 - t_{u(y)}(y \mid \theta). \end{aligned} \tag{4.7}$$

Moyennant sur tous les x, on obtient le risque moyen qui s'interprète comme une probabilité d'erreur de classement $Prob_e$

$$\begin{aligned} Prob_e(u) &= E_Y[\mathcal{R}(u \mid Y)] \\ &= 1 - E_Y[t_{u(Y)}(Y \mid \theta)]. \end{aligned} \tag{4.8}$$

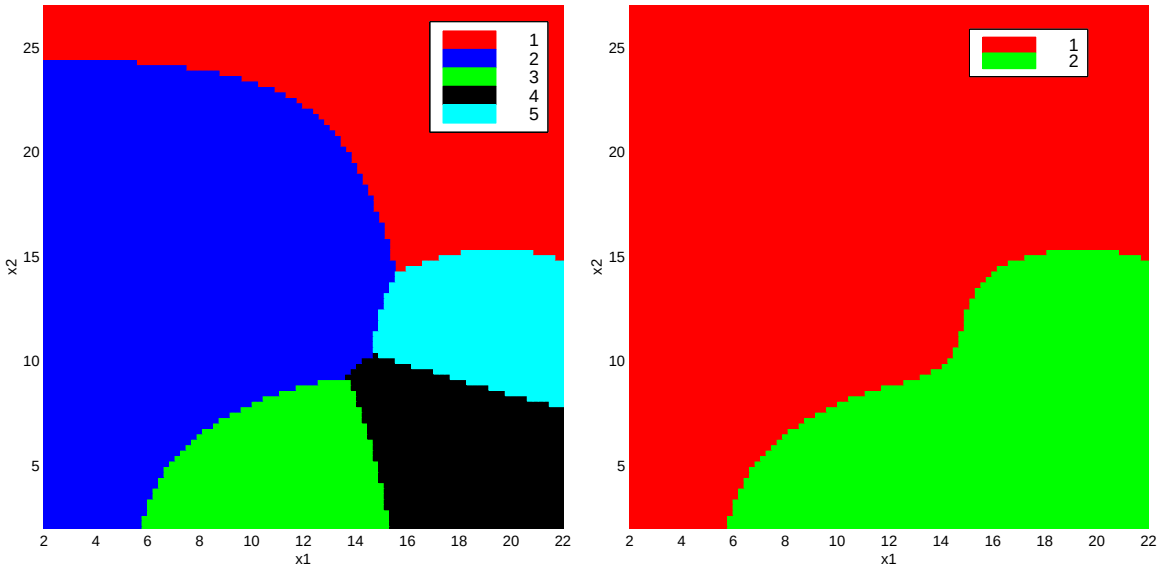


FIG. 4.3: *Illustration de la deuxième étape de l'approche proposée sur les données de la figure 4.2. Classement des points de l'espace* $\mathbb{R}^2$ *par la règle du MAP. (a) Les* 5 *zones d'appartenance aux* 5 *classes d'apparences intra-plan; Pour chaque point de la grille les probabilités a posteriori* $t_{ik}(\theta)$ *d'appartenances aux* $K = 5$ *classes d'apparences intra-plan sont calculées, et en appliquant le MAP, chaque point est classé dans la classe la plus probable; (b) Les zones d'appartenances aux deux classes d'objets d'origine déduites de l'étape précédente (a) par la formule (4.5).*

La règle de classement optimal u_0, ou règle de Bayes, est celle qui minimise l'erreur de classement $Prob_e$. Pratiquement, il suffit de minimiser le risque conditionnel $\mathcal{R}(u \mid y)$ pour chaque y. D'où la règle de MAP

$$\forall y \in \mathbb{R}^d \quad , \quad u_\theta(y) = \arg\min_k t_k(y \mid \theta).$$

Le paramètre $\theta = \hat{\theta}$ étant estimé à partir de données (X, Z), ceci permet d'estimer les probabilités a posteriori par $\hat{t}_{ik}(\theta) = t_{ik}(\hat{\theta})$ et ensuite on en déduit la règle de classement optimale $\hat{u}_\theta(y) = u_{\hat{\theta}}(y)$ (c'est le principe de la méthode du *plug-in*).

4.3.6 Classement robuste des apparences d'objets

Soit Ω_r un objet suivi requête. On cherche à lui trouver la classe la plus proche parmi les L classes d'objets suivis. Le MAP associe chaque occurrence y_i de Ω_r à une classe ℓ. Cependant, tous les y_i ne sont pas probablement associés à la même classe. Ceci est dû principalement, comme nous l'avons déjà mentionné plusieurs fois auparavant, aux variations de l'apparence de l'objet suivi au cours du temps. On se place dans l'hypothèse que seulement quelques occurrences de Ω_r ont des apparences trop éloignées du reste. La figure 3.2 illustre un exemplaire de ce cas : les quelques dernières apparences de l'enfant

qui court sont visuellement des apparences aberrantes par rapport à l'ensemble total de ses apparences dans le plan vidéo. Il en résulte alors une mauvaise classification de ces occurrences aberrantes par le MAP.

En appliquant le vote majoritaire, nous rectifions d'une façons robuste les mauvais classements des y_i de Ω_r et ensuite nous identifions la classe de Ω_r. Dans nos expérimentations réelles (section 4.4), cette technique de classement robuste a augmenté le taux de bon classement des occurrences d'objets suivis de 10% ou plus selon les cas. On note dans la suite $cr.\%$ le pourcentage de bons classements robustes des apparences d'objets.

4.4 Expérimentations

4.4.1 Séquence Avengers-1

Cette séquence de 1016 images comporte 1391 apparences d'objets qui correspondent à 52 différents objets suivis dans 31 plans. Elle est extraite du corpus vidéo fourni par l'INA (section 2.5). Cette séquence servira de base pour les expérimentations de l'approche de classification supervisée proposée dans ce chapitre. Sa particularité est que d'une part plusieurs plans ont des durées très courtes (moins d'une seconde), et d'autre part l'apparence intra-plan des objets est très variable. Toutes sortes de changements de l'image sont présentes dans cette base d'objets: occultation partielle, changement d'éclairage, bruits, changement de point de vue, rotation 3D et changement d'échelle (voir section 3.1.1). Les figures 4.4 et 4.5 illustrent quelques apparences d'objets de cette séquence.

4.4.2 Les modèles d'objets

Un utilisateur expert, peut associer les occurrences de tous les objets suivis de la séquence Avengers-1 à 12 classes d'objets différentes (voiture Ford blanche, voiture Mercedes, acteur J. Steed, actrice Perley, etc). En utilisant le système désigné pour ce travail (voir section 4.3), un modèle d'objet suivi a été sélectionné parmi chaque classe d'objets. De ces 12 modèles d'objets suivis, 448 apparences différentes ont été collectées et serviront pour l'estimation de la loi globale de mélange gaussien. La figure 4.4 illustre 6 différents modèles d'objets suivis; pour chaque modèle d'objet suivi quatres apparences intra-plan sont affichées.

4.4.3 Les données

Le choix d'une caractérisation de bas niveau à stocker sous la forme d'un vecteur de descripteurs, censé représenter une image, est crucial. Un survol des descripteurs des couleurs, des formes et des textures a été présenté dans la section 3.2. Chaque descripteur a ses propres caractéristiques et aucun ne peut être robuste à tous les changements de l'apparence des objets de notre base expérimentale. D'autre part, une restriction de notre approche est l'utilisation des descripteurs globaux uniquement; la modélisation de la variabilité est effectuée dans un espace $\mathbb{R}^d$, où d est la taille du vecteur descripteur caractérisant une apparence d'un objet suivi de la base.

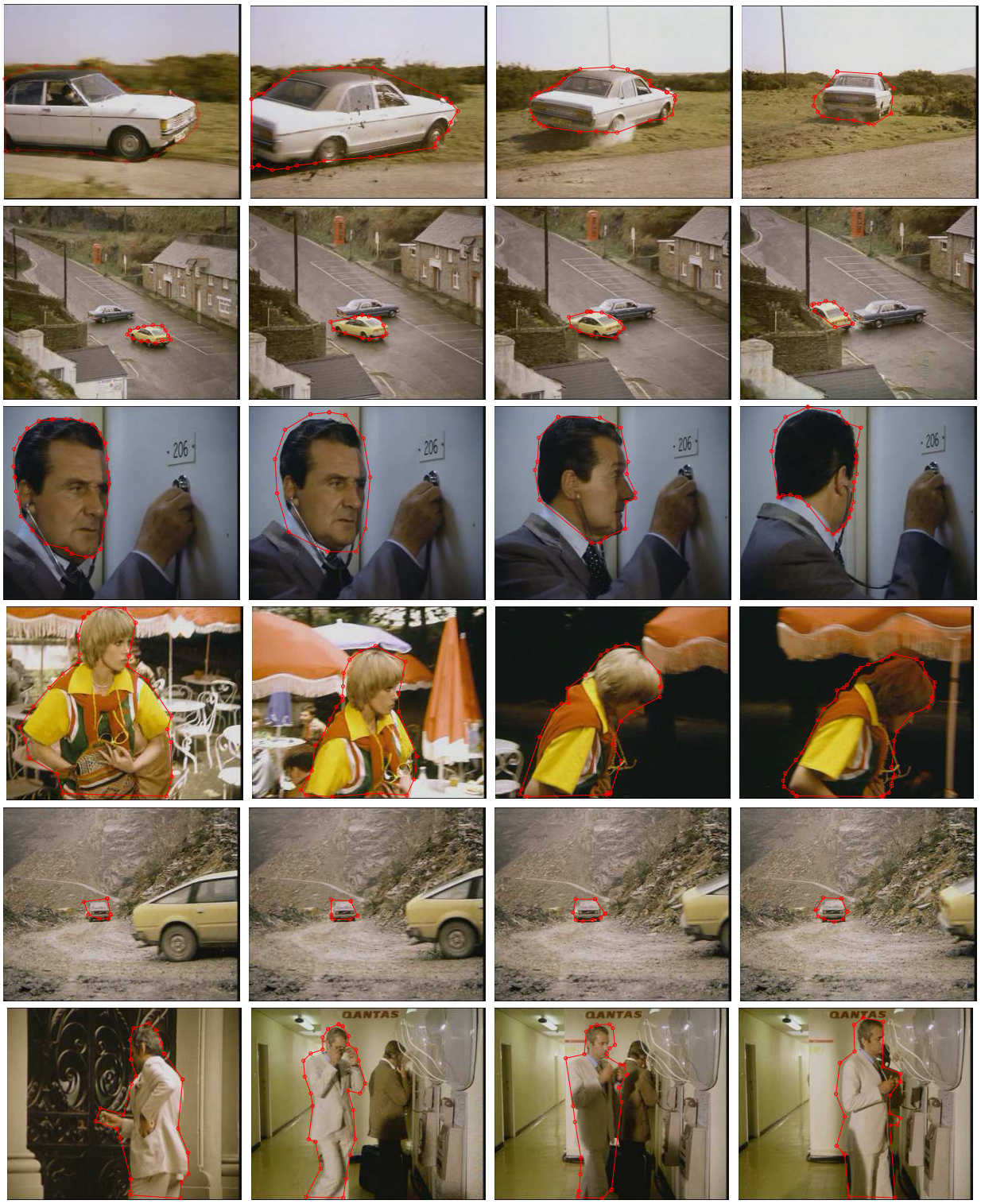


FIG. 4.4: *Quelques modèles d'objets suivis de la séquence vidéo Avengers-1.*

Vu que les objets vidéo sont mobiles et donc leurs apparences à travers le temps sont trop variables, la caractérisation de bas niveaux par des descripteurs de la forme n'est certainement pas le bon choix. Par contre, l'information de la couleur semble être une information visuellement discriminante et aussi très présente dans la base d'objets (voir figures 4.4 et 4.5). Ce choix a été discuté en détail dans la section 3.3.

L'histogramme de couleurs est calculé dans les espaces de couleurs RGB, HSV , HS, H (tinte), S (saturation), I (niveaux de gris), rgb (chromaticité) et $l_1l_2l_3$. Afin de prendre en compte le nombre réduit des apparences des modèles d'objets suivis, les espaces de couleurs sont quantifiés en un nombre de couleurs raisonnable mais suffisament discriminant: les espaces RGB, HSV, rgb, $l_1l_2l_3$, HS, H, S, et I sont quantifiés en 64, 64, 64, 49, 32, 32, 32 et 32 couleurs respectivement. En plus, on considère que les axes d'un histogramme disposent d'un nombre de cellules équivalent (par exemple 4 cellules sur les 3 axes R, G et B de l'espace RGB). Notons que la quantification de l'espace de couleurs est faite d'une façon empirique et que d'autres dimensions peuvent être considérées [127]. L'étude de l'effet de la variation de la dimension de l'histogramme sur l'appariement d'images sort du cadre de ce travail.

Pour chaque apparence de la base des objets suivis (de toute la séquence vidéo) on extrait les histogrammes cités ci-dessus. Ensuite, sur chaque tableau de données $n \times d$ ainsi obtenu pour les n apparences d'objets de la séquence vidéo traitée, où chaque ligne de ce tableau est un histogramme de d dimensions, l'analyse en composante principale est appliquée (voir section 3.4).

Ceci permet de projeter les histogrammes RGB, HSV, rgb, $l_1l_2l_3$, HS, H, S et I de la séquence Avengers-1 dans les espaces à $10, 10, 3, 10, 8, 5, 8$ et 8 dimensions respectivement. Une qualité de représentation Q_E (formule 3.5) des données dans ces espaces réduits de 95% est fixée a priori. Dans la suite, on note par d et d_E les dimensions de l'espace de descripteurs initial et réduit respectivement.

4.4.4 Paramètrage de l'approche

Dans un premier temps, les classes d'apparences intra-plan de chaque modèle d'objet suivi sont identifiées. Ces classes sont obtenues par une modélisation de la variabilité de l'objet suivi dans l'espace de descripteurs $\mathbb{R}^{d_E}$, par un mélange gaussien. Le nombre de composantes gaussiennes maximal, noté $MaxNbC$, peut varier de 1 à 4. Trois critères, BIC, ICL et NEC sont utilisés dans les expérimentations pour sélectionner automatiquement la meilleure structure de la loi de mélange (voir section 3.6.6).

Les 12 lois de mélanges des modèles d'objets suivis sont ensuite réunies sous la forme d'une seule loi globale de mélange gaussien.

4.4.5 Les requêtes

La loi globale de mélange gaussien ainsi obtenue pour les 12 modèles d'objets suivis permet au système de classer toute nouvelle apparence d'un objet suivi (requête individuelle) dans la vidéo Avengers-1 dans une classe unique (voir section 4.3.5). Dans un premier test, 943 nouvelles apparences différentes qui ne sont pas auparavant utilisées

FIG. 4.5: *Quelques apparences requêtes de la séquence Avengers-1.*

dans l'apprentissage de la loi de mélange global sont classées par le système. La figure 4.5 illustre quelques apparences d'objets requêtes. Aussi, les 448 apparences utilisées dans l'apprentissage sont aussi classées, afin de tester l'effet de la fusion des lois de mélanges des modèles d'objets suivis dans une même loi globale. Rappelons que dans ce cas chaque apparence d'un objet suivi est classée indépendamment des autres apparences du même objet suivi. Dans un deuxième test, les 52 objets suivis sont considérés eux même comme des requêtes où toutes les apparences d'un même objet suivi sont rangées dans la même classe : approche de classement robuste (voir section 4.3.6).

4.4.6 Les résultats

Une comparaison avec les classements manuels basés sur la vision humaine semble être le seul moyen pour évaluer les résultats de l'approche proposée. En adoptant cette démarche, les pourcentages $ci.\%$ et $cr.\%$ de bon classement individuel et robuste sont comptés pour la totalité des requêtes (voir section 4.3.5 et 4.3.6).

Les résultats de classement sont rangés dans les tableaux 4.1 et 4.2. Le premier rang indique l'espace de descripteurs sur lequel la loi globale de mélange gaussien est construite. Les dimensions initiales et réduites de cet espace sont indiquées dans le deuxième et troisième rang respectivement. Les pourcentages $ci.\%$ et $cr.\%$ sont montrés dans le cas où les critères BIC, ICL et NEC sont utilisés. Une illustration graphique de ces tableaux est donnée dans les figures 4.8 et 4.9.

L'analyse de ces tableaux permet de tirer plusieurs conclusions très intéressantes des différents aspects de l'approche :

- Les **meilleurs pourcentages** obtenus avec les approches de classement individuels et robustes des apparences d'objets sont de 81.5% et 90.2% respectivement. La modélisation de la variabilité intra-plan des objets suivis est effectuée dans ce cas dans l'espace de descripteurs RGB à 10 dimensions, où le nombre maximal de classes d'apparences intra-plan ($MaxNbC$) est égal à 2 et où les critères BIC et ICL sont employés. Ces scores sont très satisfaisants vu la particularité de la base d'objets de la séquence Avengers-1 (voir section 4.4.1). Les figures 4.6 et 4.7 montrent quelques résultats de classement des requêtes issues de cette expérience.

- "**Mieux** la variabilité est modélisée – en terme du nombre de composantes gaussiennes et de la complexité du modèle gaussien sélectionné – **mieux** les frontières des classes d'objets sont estimées et donc **plus** les classements sont corrects". Cette règle est validée expérimentalement, dans presque tous les espaces de descripteurs testés, lorsque le nombre maximum de composantes gaussiennes ($MaxNbC$) progresse de 1 à 2; ainsi on observe une amélioration des pourcentages $ci.\%$ et $cr.\%$ entre 20% (exemple de $cr.\%$ dans l'espace $l_1l_2l_3$ avec BIC) et 4%. Par contre, ces résultats commencent à se dégrader quand $MaxNbC$ dépasse 2, dans les espaces de descripteurs RGB, HSV et $l_1l_2l_3$, ce qui n'est pas étonnant car le nombre des individus qui ont participé à l'apprentissage n'est pas assez suffisant pour estimer des frontières optimales entres les classes d'objets : la distribution d'un objet suivi parait multimodale, ainsi avec un nombre des apparences assez petit par rapport à la taille

FIG. 4.6: *Résultats de classement de quelques apparences requêtes dans les modèles d'objets suivis de la séquence vidéo Avengers-1. Toutes ces requêtes sont correctement classées.*

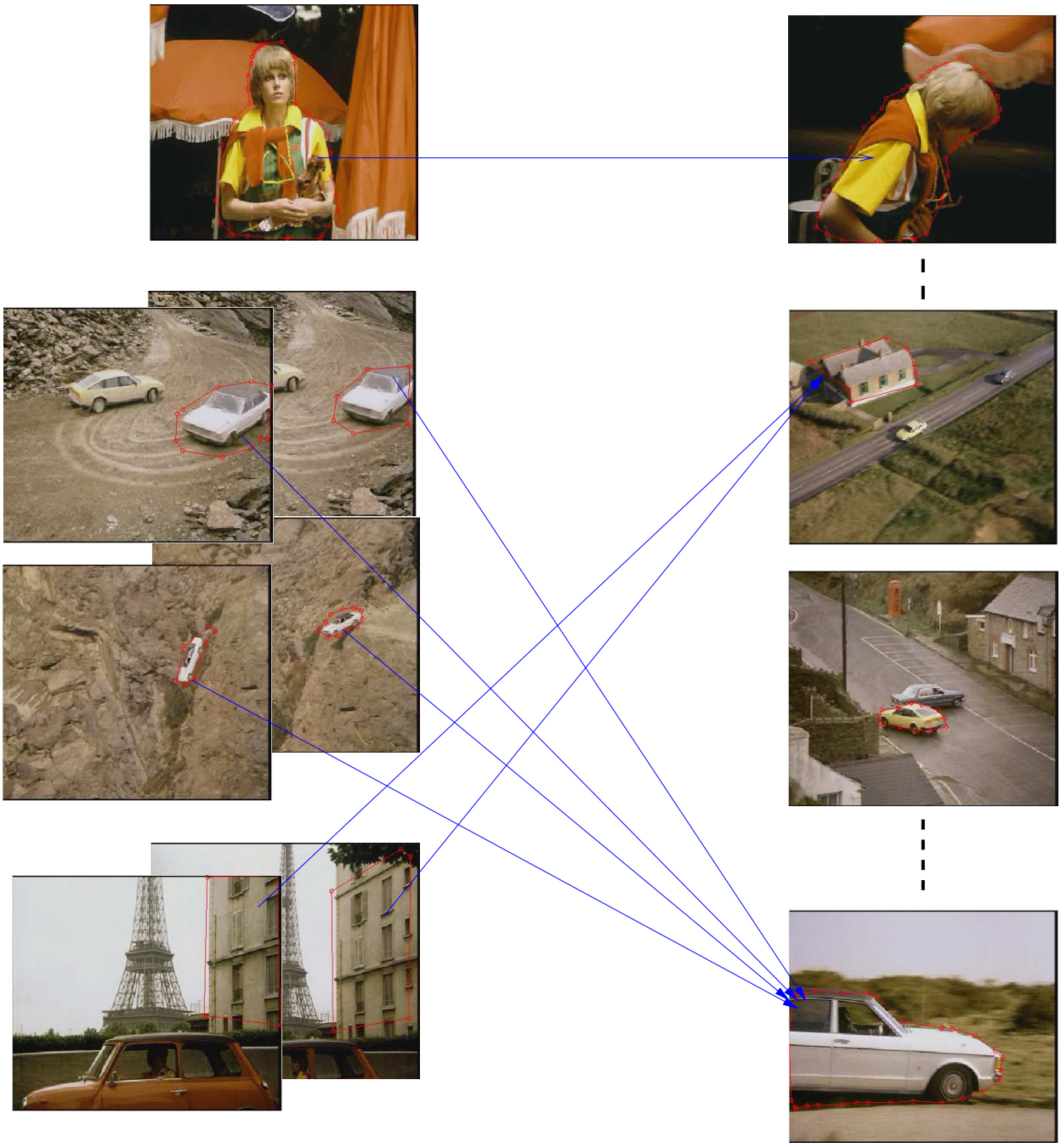


FIG. 4.7: *Résultats de classement de quelques apparences requêtes dans les modèles d'objets suivis de la séquence vidéo Avengers-1. Toutes ces requêtes sont correctement classées.*

de ces espaces les critères utilisés ont souvent tendance à choisir un nombre de classes assez grand, mais avec des modèles simples. Dans des espaces de dimensions plus petites la validité de la règle ci-dessus est manifestement plus nette. Par exemple, dans les espace HS, H, S et I à 8 dimensions, on trouve que les pourcentages de bons classements sont les plus élevés quand $MaxNbC$ est égal à 4. On a $cr.\%$ est égal à 71.9%, 63.3% et 61.8% dans les espaces HS, I et H respectivement. Le critère ICL a été utilisé dans ce cas.

Espace de descripteurs	d	d_E	$MaxNbC$	$ci.\%$			$cr.\%$		
				BIC	ICL	NEC	BIC	ICL	NEC
RGB	64	10	1	77.9	77.9	43.7	85.3	85.3	46.8
RGB	-	-	2	81.0	81.5	57.9	90.2	89.1	58.9
RGB	-	-	3	81.0	80.3	65.7	86.5	79.9	69.3
RGB	-	-	4	77.2	79.9	57.8	82.5	86.8	62.3
HSV	64	10	1	67.6	67.6	44.4	68.3	68.3	38.9
HSV	-	-	2	70.2	67.4	41.6	77.8	72.3	39.1
HSV	-	-	3	66.5	69.1	43.5	71.7	75.8	43.7
HSV	-	-	4	66.4	66.7	59.2	68.6	71.7	56.7
HS	49	8	1	60.2	60.2	44.2	54.0	54.4	44.6
HS	-	-	2	55.7	55.6	50.4	64.0	59.2	45.1
HS	-	-	3	53.6	52.0	68.6	55.0	52.9	67.6
HS	-	-	4	52.6	67.0	63.5	50.3	71.9	62.4
I	32	8	1	42.8	42.8	42.8	45.4	45.8	45.4
I	-	-	2	49.4	51.0	48.7	52.0	51.0	48.6
I	-	-	3	52.0	51.2	51.7	52.4	51.0	55.4
I	-	-	4	52.0	52.0	48.6	54.1	63.3	51.0

TAB. 4.1: *Résultats des approches de classement individuels et robustes des apparences d'objets de la séquence vidéo Avengers-1, dans les espaces de descripteurs RGB, HSV, HS et I.*

4.5 Approches classiques de reconnaissance des objets vidéo

Une approche classique d'indexation des objets d'une séquence vidéo consiste à indexer seulement les objets qui se trouvent dans les images clés [92]. Cette approche est considérée comme une solution simplificatrice du problème de gestion de la taille gigantesque de la vidéo (voir l'introduction du chapitre précédent), en ignorant le problème de la variation intra-plan du contenu. Notons que cette approche a été implémentée dans la première version de notre système de construction de la vidéo hyperliée (voir chapitre 8). L'image médiane est souvent prise comme image représentative du plan. Plus tard, Zhang [133] propose la méthode de "moyenne temporelle de descripteurs" (appelé *temporal mean*

Espace de descripteurs	d	d_E	$MaxNbC$	$ci.\%$			$cr.\%$		
				BIC	ICL	NEC	BIC	ICL	NEC
H	32	5	1	54.5	54.5	38.0	54.7	54.7	43.1
H	-	-	2	60.5	54.7	59.2	61.8	52.1	65.8
H	-	-	3	57.1	50.5	45.1	59.2	50.7	45.9
H	-	-	4	58.2	60.5	44.8	62.9	61.8	47.4
S	32	8	1	48.5	48.5	39.5	47.1	47.1	39.0
S	-	-	2	59.5	50.8	50.8	63.0	56.6	50.2
S	-	-	3	45.1	51.3	48.8	45.6	55.9	51.3
S	-	-	4	53.1	49.5	46.2	52.6	55.2	45.4
rgb	64	3	1	36.2	36.2	33.8	40.3	40.3	36.7
rgb	-	-	2	38.5	37.2	34.8	38.5	40.0	39.3
rgb	-	-	3	40.5	37.4	35.1	41.4	41.3	36.7
rgb	-	-	4	38.1	38.6	36.0	40.8	41.7	38.9
$l_1l_2l_3$	64	10	1	43.0	43.0	39.8	41.8	41.8	42.3
$l_1l_2l_3$	-	-	2	56.6	54.6	38.2	61.3	61.5	38.0
$l_1l_2l_3$	-	-	3	55.0	55.8	43.3	58.7	60.5	46.2
$l_1l_2l_3$	-	-	4	51.3	52.4	41.0	55.6	53.7	41.3

TAB. 4.2: *Résultats des approches de classement individuels et robustes des apparences d'objets de la séquence vidéo Avengers-1, dans les espaces de descripteurs H, S, rgb et $l_1l_2l_3$.*

feature en anglais) qui intègre l'aspect temporel dans le calcul de la similarité entre les plans vidéo; il calcule la moyenne de la totalité des descripteurs extraits des images du plan, en particulier il extrait la luminosité et quelques couleurs dominantes dans l'image.

Avec comme objectif la comparaison de performance de l'approche proposée et la méthode de Zhang, nous avons testé cette dernière sur la base d'objets de la séquence Avengers-1. Effectivement, la méthode de Zhang est un cas particulier de notre approche, où chaque modèle d'objet suivi est représenté par le centroïde μ de la distribution avec une variance nulle, commune pour tous les modèles d'objets suivis. Dans la base d'indexes, il y aura donc seulement 12 descripteurs. Le jeu de requête utilisé auparavant est considéré dans l'évaluation de cette approche classique. D'abord, une requête est appariée avec la base par le calcul d'une distance euclidienne. Ensuite, elle est classée par la méthode du premier proche voisin. Et enfin, un classement robuste est appliqué. Le tableau 4.3 décrit les résultats de test de cette approche, dans le cas de classement individuel et robuste, et ceci pour tous les espaces de descripteurs à dimensions réduites.

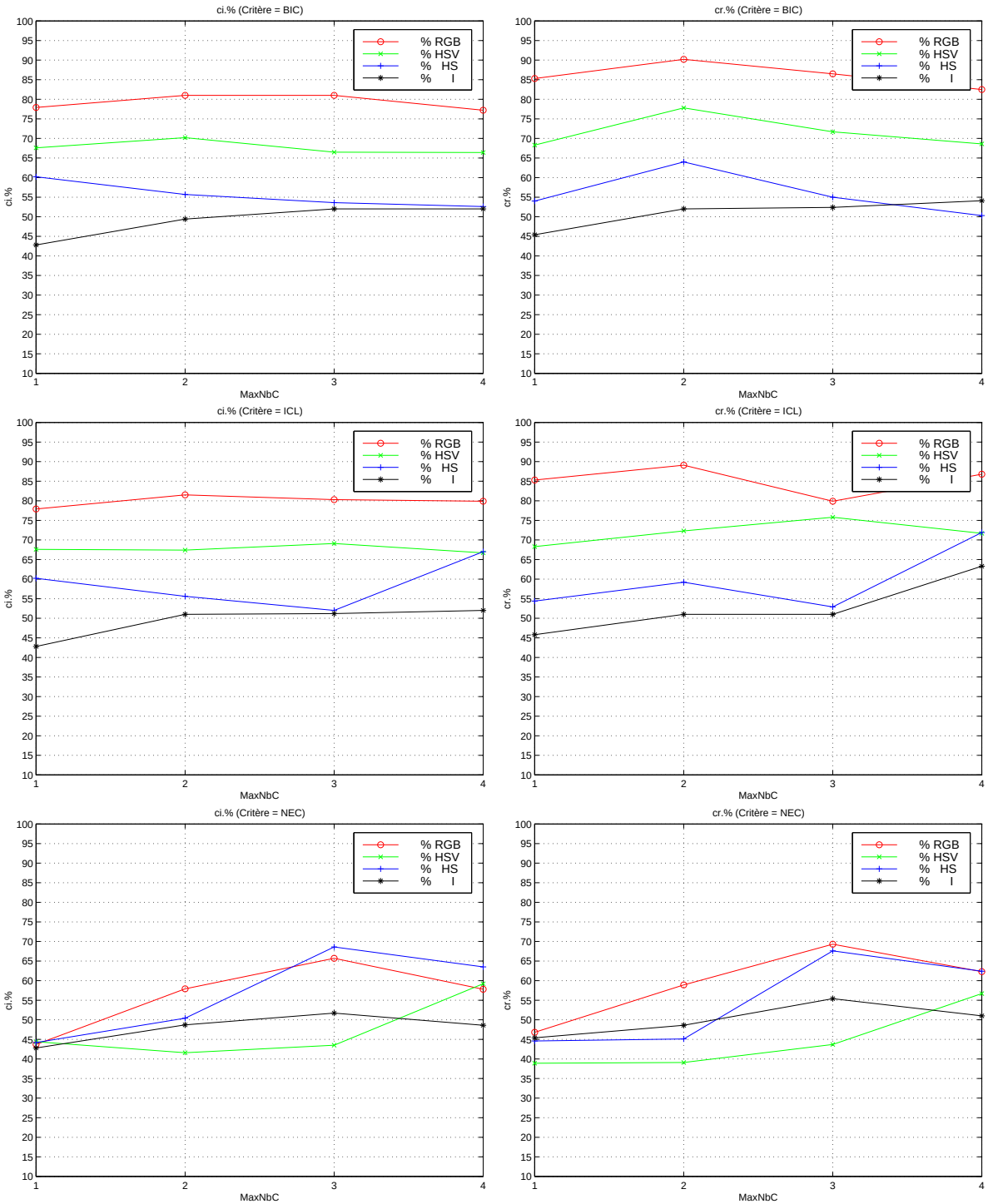


FIG. 4.8: *Illustration graphique des résultats des approches de classement individuels (colonne gauche) et robustes (colonne droite) des apparences d'objets de la séquence vidéo Avengers-1; Les résultats avec les critères BIC, ICL et NEC sont affichés dans le premier, deuxième et dernier rang respectivement.*

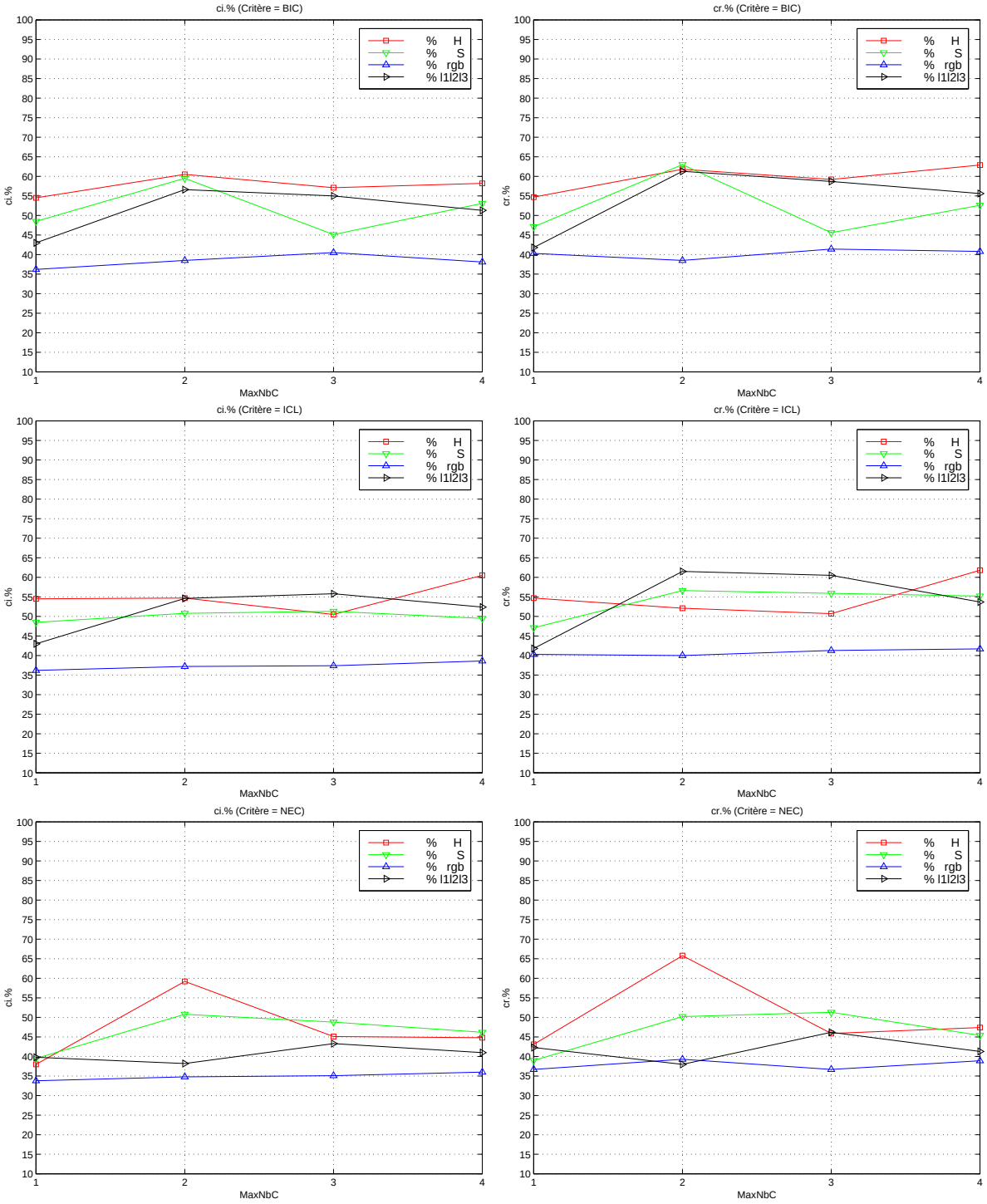


FIG. 4.9: *Suite de la figure 4.8*

Descripteur	d_E	*ci.*%	*cr.*%
RGB	10	56.6	55.3
HSV	10	53.7	60.3
HS	8	48.2	48.4
I	8	36.6	41.1
H	5	42.4	48.7
S	8	39.5	39.5
rgb	3	35.4	41.3
$l_1l_2l_3$	10	42.4	42.7

TAB. 4.3: *Résultats de scores de bon classement par l'approche classique de la "moyenne temporelle de descripteurs".*

4.6 Analyse comparative et discussion

Performance de l'approche proposée. L'approche de classification supervisée proposée dans ce chapitre pour classifier les objets suivis en groupes homogènes est basée sur une idée fondamentale : prendre en compte la variation de l'apparence intra-plan des modèles d'objets suivis. L'intégration de cet aspect temporel consiste à identifier les classes d'apparences intra-plan des objets suivis indépendamment par des lois de mélanges gaussiens. Le nombre de classes ainsi que le modèle gaussien adaptés à une distribution d'un objet suivi dans l'espace de descripteurs sont déterminés à l'aide des critères probabilistes. Même si l'objet suivi n'est pas mobile dans le plan vidéo, souvent sa distribution dans l'espace de descripteurs n'est pas compacte autour d'un centre et avec une variance (presque) nulle pour les raisons citées dans l'introduction de ce chapitre. Il en résulte que la méthode de Zhang (*moyenne temporelle de descripteurs*) semble être inefficace pour indexer les objets vidéo. Par contre, notre approche est en quelque sorte une extension de cette approche classique où une distribution est représentée par un ou plusieurs centres avec des variances autour de ces centres. On s'attend donc à ce qu'elle soit beaucoup plus performante; les expérimentations ci-dessus montrent une amélioration significative des pourcentages de classements robustes de 10% à 35% par notre approche. La figure 4.10 illustre les performances de deux approches.

Classement robuste. L'approche de classement robuste améliore les résultats de classement individuel des apparences d'objets jusqu'à 10%. Ceci est un peu faible. Lorsqu'un peu d'apparences d'un objet suivi sont aberrantes et que la majorité est bien classée, alors le classement robuste rectifie les mauvais classements. Dans ce cas, les pourcentages de bon classements vont progresser et, dans le cas inverse, ils vont décroître. Sur la base d'objets ainsi expérimentée, ces deux situations sont malheureusement présentes, l'une pénalisant l'autre. Par contre, le premier cas est plus présent car les pourcentages ont augmenté.

Le deuxième cas, où la majorité des apparences d'un objet suivi est mal classée, peut

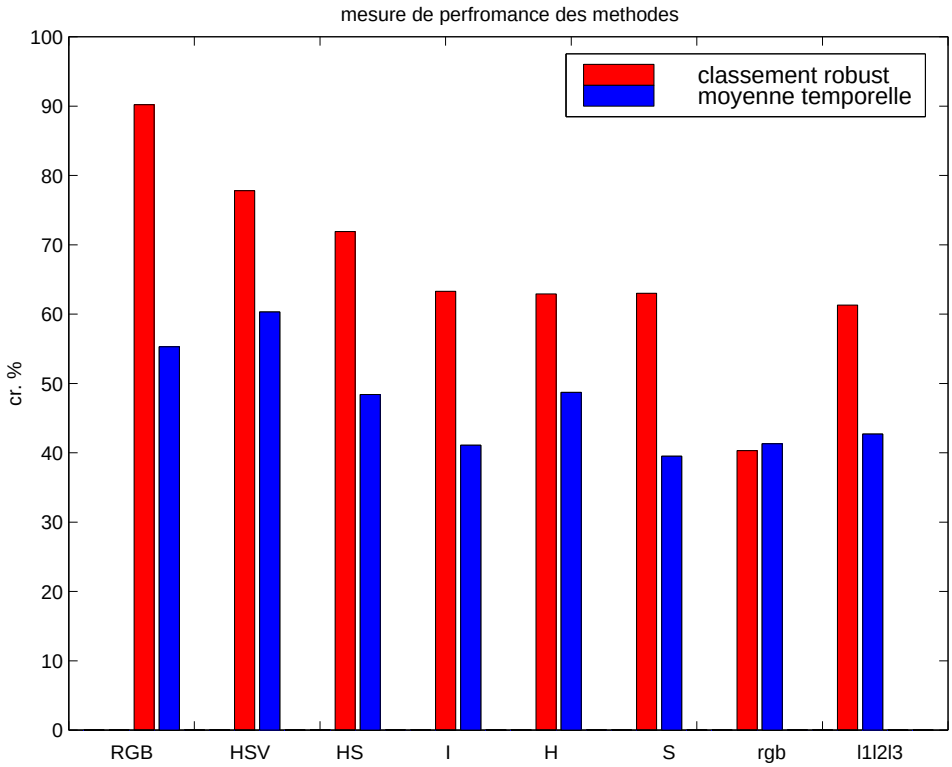


FIG. 4.10: *Comparaison des résultats de l'approche de classement robuste et celui de la "moyenne temporelle de descripteurs".*

être expliqué par le fait que ces apparences sont trop éloignées (visuellement et dans l'espace de descripteurs) du modèle d'objet suivi. Notons que, dans ce travail, on suppose que pour chaque requête existe un modèle d'objet. Par contre, un modèle d'objet suivi est sélectionné dans le plan vidéo et parfois ses apparences ne recouvrent entièrement pas toutes les apparences dans les autres plans. Il serait en de posséder d'un modèle 3D du modèle d'objet d'où l'on pourrait produire des apparences virtuelles multiples à savoir le modèle convenable de transformation du modèle d'objet. Cela permettrait de résoudre le problème des données limitées pour l'estimation.

D'autre part, si les modèles d'objets suivis ne sont pas représentatifs (aux moins dans le sens sémantique) de toutes les requêtes, un processus de rejet est indispensable pour le système. On peut utiliser la loi de χ^2 pour rejeter toute apparence déjà classée dans k par la règle de MAP, qui a une distance de Mahalanobis à cette classe k qui n'est pas dans un intervalle de confiance de 95% par exemple. Du point de vu technique, le rejet n'a aucun sens; des objets dans la vidéo ne font pas partie des classes d'objets ainsi fabriquées par le système dont l'objectif est la création de la vidéo hyperliée: un objet point sur toutes ces occurrences dans la vidéo.

Notons que le changement de la base des modèles d'objets suivis peut influencer relativement les résultats. Dans [57], nous avons changé légèrement la base d'objets en utilisant

15 modèles d'objets suivis et des changements mineurs des résultats ont été remarqués.

Structure de la loi de mélange. Le choix d'un critère pour déterminer la structure d'un mélange gaussien a un effet significatif sur les résultats finaux de classement des requêtes. Lorsque la distribution de chaque objet suivi est modélisée par une seule composante gaussienne ($MaxNbC = 1$) le critère NEC semble être mal adapté pour le choix du modèle gaussien le plus discriminant (frontière optimale) parmi les 7 modèles gaussiens mis en compétition. C'est la seule explication des mauvais résultats obtenus lorsque ce critère est employé. Par contre BIC et ICL donnent des meilleurs résultats par la sélection du modèle gaussien le plus complexe.

Les tests prouvent que dans les espaces de descripteurs RGB, HSV, $l_1l_2l_3$ et rgb la modélisation des distributions des modèles d'objets suivis par des lois de mélanges de deux composantes gaussiennes au maximum donne les meilleurs résultats. Ceci dit que avec un nombre de composantes plus élevé le risque de mal adaptation (*over-fitting* en anglais) des données était aussi très grand; l'estimation des paramètres gaussiens est devenue instable car le nombre des individus par classe gaussienne est trop petit par rapport à la dimension de l'espace d'apprentissage. Par contre, dans les espaces de dimension plus petite I, H, S, et HS, la modélisation avec 4 gaussiennes au maximum donne les meilleurs résultats de classement. Le critère ICL semble être le mieux adapté dans ce cas.

Descripteurs invariants aux changements de luminosité. La méthode de classement fournit des très bons résultats dans l'espace de descripteurs RGB et de très mauvais résultats dans l'espace rgb normalisé. Idem, avec la méthode classique de Zhang. C'est surprenant vis à vis des résultats obtenus par Finlayson [36] et Gevers [43]. Mais, deux choses expliquent ce phénomène : (1) un histogramme rgb quantifié en 64 couleurs contient plus de deux tiers de cases vides (fréquences nulles), car les couleurs normalisées de pixels sont centrées autour de zéro; (2) ensuite, le fait d'appliquer une méthode linéaire comme l'ACP sur un ensemble des histogrammes rgb réduit la dimension de l'espace de représentation de ces données du 64 à 3, avec une qualité de représentation de 95%; dans ce nouvel espace la représentation des apparences d'objets n'est certainement plus assez discriminante et donc il y aura beaucoup de chevauchements entre les densités de mélange gaussien des différents modèles d'objets suivis, ce qui explique les classements incorrects des requêtes. Aussi, on peut rajouter à ce qui précède que la base de tests contient des changements de luminosité naturelle (soleil, ombrage, etc., voir figure 4.5) et qui ne correspondent généralement pas aux modèles théoriques implémentés ici. Également, les changements de luminosité sont appliqués partiellement sur les objets, ce qui rend critique l'application de la normalisation de couleurs. En effet, ce type de changements de luminosité devra peut être pris en compte par les chercheurs qui travaillent sur ce sujet de normalisation de la couleurs, qui sort du cadre de cette thèse. Enfin, l'invariance repose sur un modèle théorique qui supprime brutalement une information significative : sous prétexte de supprimer l'incidence d'une variabilité, somme toute limitée, de la luminance, on supprime complètement cette information.

Une solution pourrait être envisagée : considérer des histogrammes de couleurs rgb de

dimension plus élevée (4096 classes de couleurs par exemple). Cependant, ceci est loin d'être validé dans nos expérimentations car le nombre des apparences est très limité et nos expérimentations montrent qu'une telle situation conduit à une estimation instable.

Enfin, rappelons que la segmentation d'objets (durant la phase de suivi) n'est pas fiable et elle contribue à la variabilité des objets et ensuite aux mauvais classements des requêtes.

Descripteur	d	$ci.\%$	$cr.\%$
RGB	64	61.4	59.3
HSV	64	59.0	60.0
HS	49	52.9	56.6
I	32	37.8	43.2
H	32	53.7	56.1
S	32	42.3	40.8
rgb	64	30.4	25.6
$l_1l_2l_3$	64	45.5	48.3

TAB. 4.4: *Résultats des scores de bon classement par la méthode de "moyenne temporelle de descripteurs" dans l'espace initial de descripteurs.*

Projection des données. Il est très difficile de conserver la même variance de données lors d'une projection dans un espace de petite dimension (section 3.4). L'analyse en composante principale ne prend pas en compte la structure non linéaire des données, des structures contenant des groupes ayant des formes arbitraires. Visuellement, il est impossible de vérifier si nos données ont des structures particulières dans les espaces à 64 dimensions, mais elles ne sont probablement pas linéaires. Ceci peut être validé par exemple par le fait que les pourcentages de bons classements obtenus dans l'espace initial de descripteurs avec la méthode de la "moyenne temporelle de descripteurs" et en utilisant la distance de χ^2 (formule 3.3) normalisée pour l'appariement sont plus élevés que dans l'espace réduit (voir tableaux 4.4 et 4.3).

Par contre, une méthode non-linéaire comme l'analyse en composante curvilignes (ACC) [30] et la méthode *multidimensionnel scaling* (MDS) [73] peuvent être employées. Cependant, des tests menés dans notre équipe ont montrés que l'ACC ne donne pas des meilleurs résultats que l'ACP [33]. D'autre part, la méthode MDS est coûteuse et n'est pas bien adaptée pour des données de taille importante (par exemple il nous a fallu plus de 3 heures pour une matrice de données de taille 1391×64 lorsque l'optimisation de la fonction de coût converge vers une solution).

4.7 Conclusion

Ce chapitre a présenté une approche de classification supervisée des objets suivis, où l'auteur de la vidéo hyperliée intervient dans la sélection des "modèles d'objets suivis" pour

une séquence vidéo traitée. Le mélange gaussien est employé dans ce cas de discrimination. Mais son utilisation n'est pas classique ici, car chaque "modèle d'objet suivi" (une classe en discrimination) est lui même représenté par un mélange de modèles gaussiens : les classes d'apparences intra-plan modélisées par des gaussiennes. Rappelons que la structure de ces classes d'apparences intra-plan est déterminée automatiquement comme nous l'avons vu dans le chapitre précédent.

Après l'identification de ces classes d'apparences intra-plan de tous les modèles d'objets suivis, une loi globale de mélange gaussien est construite par un nouveau calcul des paramètres des proportions seulement. Le classement des nouvelles apparences d'objets dans la vidéo est effectué par deux méthodes : classement par maximum a posteriori et classement robuste par vote majoritaire.

Les expérimentations menées ici sur huit types de descripteurs globaux (histogrammes de couleurs bruts et normalisés) permettent de tirer les conclusions suivantes :

- La méthode de classement robuste donne de meilleurs résultats d'un ordre de 10% environ que celle du maximum a posteriori.
- La méthode de classement robuste donne des résultats bien meilleur d'un ordre allant jusqu'à 35% que la méthode classique de la "moyenne temporelle de descripteurs".
- Les résultats sont du même ordre quand les deux critères BIC et ICL sont utilisés pour le choix automatique de la structure des classes d'apparences intra-plan. Par contre lorsque le critère NEC est utilisé les résultats sont très mauvais. Ce critère n'est pas adapté au choix du modèle gaussien.
- Les résultats sont les meilleurs lorsque les classes d'apparences intra-plan sont recherchées dans l'espace des histogrammes de couleurs RGB réduit par ACP. La réduction de l'espace des histogrammes normalisés rgb conduit à une perte d'information significative ($d_E = 3$ avec une qualité de représentation des données de 95%) et ensuite à un chevauchement très large entre les classes d'apparences intra-plan des différents modèles d'objets suivis. Ceci explique les mauvais résultats de classement obtenus dans cet espace. Une telle réduction doit être évitée.

Une méthode alternative pour palier le problème de données limitées, et donc pour éviter la réduction de l'espace par ACP est de générer peut-être des apparences virtuelles des modèles d'objets suivis. La question qui se pose à ce stade : quel modèle mathématique faut-il appliquer pour générer des apparences virtuelles ? un réponse simple possible est la fabrication automatique d'occultation partielle. Une autre méthode qui nous paraît naturelle est d'effectuer l'apprentissage d'une classe d'objets sur plusieurs apparences d'un même objet suivi dans différents plans.

Chapitre 5

Classification automatique des objets vidéo

Chaque jour, me fixer 5 actions prioritaires.

5.1 Introduction

Dans le contexte des problématiques et des motivations des deux chapitres précédents, nous présentons dans ce chapitre une approche de classification automatique des objets suivis. La motivation plus particulière que nous avons ici est de répondre au besoin suivant : considérons un utilisateur qui doit identifier puis répertorier tous les objets d'une vidéo et les grouper dans des classes; si un outil permet d'opérer dans un premier temps une construction approximative de ces classes, l'utilisateur n'aura plus qu'à les éditer plus finement, phase pour laquelle une aide pourrait encore être proposée (mais que nous n'explorons pas ici). Les données sont les paramètres gaussiens de toutes les classes d'apparences intra-plan des objets suivis, estimés dans l'espace de descripteurs classiques – l'histogramme de couleurs par exemple. La mise en correspondance entre deux objets suivis est effectuée dans cet espace de paramètres, et le but à atteindre est d'estimer la partition optimale de ces données, c'est-à-dire les classes d'équivalences inter-plans d'objets suivis. Le critère de partitionnement classique que l'on utilise est que chaque classe doit rassembler des objets aussi similaires que possible et que les classes doivent aussi être distinctes que possible les unes des autres.

5.1.1 Quelques points techniques à résoudre

Pour pouvoir classifier automatiquement les objets, des réponses satisfaisantes doivent être apportées aux questions suivantes :

1. Comment apparier deux objets modélisés par des mélanges gaussiens (ayant des structures différentes) ? Faut-il calculer une distance globale entre les densités de

mélanges gaussiens ou bien des distances individuelles entres leurs composantes gaussiennes (les classes d'apparences intra-plan des objets suivis)?

2. Quelle technique de classification automatique est applicable sur ces données? Si chaque objet suivi est représenté par un ou plusieurs points multidimensionnels dans l'espace de paramètres gaussiens, l'algorithme EM est-il applicable sur cette nouvelle distribution? Une classification hiérarchique pourra-t-elle être appliquée avec succès et un coût minimal pour choisir le meilleur nombre de classes d'objets suivis?

5.1.2 Solution adoptée

Les problèmes abordés ci-dessus seront analysés en détail plus tard. L'approche proposée dans ce chapitre consiste dans un premier temps à retenir comme distance entre deux objets suivis une distance - de Kullback , ou de Bhattacharyya - minimale entre leurs composantes gaussiennes. Ceci est justifié par le fait que deux objets suivis provenant de la même classe doivent avoir aux moins deux classes d'apparences intra-plan semblables i.e. deux composantes gaussiennes qui peuvent être identifiées. Dans un deuxième temps, la classification ascendante hiérarchique est appliquée, pour fournir une suite de partitions emboîtées en se basant sur la matrice de proximités calculée entre les objets suivis. Lors de la construction d'une hiérarchie le choix d'une mesure (indice d'agrégation) appropriée entre les classes formées et la détermination du nombre de classes restent deux handicaps. Nous fournissons à l'auteur de la vidéo hyperliée une technique interactive pour sélectionner le nombre de classes, en mettant en service des outils graphiques et visuels pour juger la qualité de la classification et pour corriger manuellement les résultats.

5.1.3 Organisation du chapitre

Après une présentation de l'état de l'art dans la section 5.2, la section 5.3 décrit en détail notre approche de classification hiérarchique des objets suivis. L'expérimentation est réalisée sur la séquence vidéo "Avengers-2" (section 2.5) de 1938 images et de 2749 apparences d'objets correspondant à 29 objets suivis différents. Cette partie sera décrite dans la section 5.4. Le choix du nombre de classes est effectué d'une manière interactive par l'utilisateur expert. La section 5.4.3 décrit l'algorithme que nous avons proposé pour évaluer les résultats expérimentaux obtenus sur la base d'objets de "Avengers-2". Cette évaluation montre un taux de bonne classification automatique autour de 80%, un pourcentage qui est bien acceptable pour une initialisation qui serait reprise par l'auteur de la vidéo hyperliée. Ce taux est aussi satisfaisant vu la grande variation de l'apparition inter-plans des objets de la même classe sémantique. Une telle technique est une solution de l'approche de classification supervisée que nous avons discutée dans le chapitre précédent : l'interaction de l'utilisateur se limite au choix du nombre de classes à la place de la sélection manuelle des modèles d'objets suivis. L'interprétation des résultats ainsi qu'une étude comparative seront présentées dans la section 5.4.4. La section 5.5 conclut ce chapitre.

5.2 État de l'art

Du fait que la classification elle-même est un domaine de recherche très ancien, motivé par les divers domaines d'applications (biologie, finance, imagerie médicale, vidéo numérique, ...), plusieurs classificateurs ont été testés en particulier sur la catégorisation d'images [32] [96] [102] : algorithme de centres mobiles, K-plus proches voisins, arbre de décision, réseaux de neurones, mélanges gaussiens, classification hiérarchique, etc.

Pun et Squire [96] fabriquent des index qui pointent sur des classes d'images, et les représentent selon une arborescente avec les images aux feuilles. L'arbre peut aussi être vu comme un arbre de décision pour la recherche d'images à partir des descripteurs. La composition des classes se fait par une analyse hiérarchique ascendante classique.

Carson et al. [20] proposent une nouvelle représentation des images. Chaque image est décomposée en plusieurs régions nommées *blobs*; chaque blob est cohérent dans l'espace de couleurs et de textures. Tous les blobs des données d'apprentissage provenant de 14 catégories d'images sont classifiés dans 180 blobs "canoniques" en utilisant le modèle gaussien diagonal. Un vecteur de scores est associé à chaque image mesurant sa similarité avec chaque blob canonique. Ensuite, ces vecteurs de scores sont utilisés pour apprendre le classificateur de l'arbre de décision. Sur la même base d'images, l'arbre de décision est aussi testé sur l'histogramme de couleurs au lieu de blobs. Comme il est mentionné dans [62] (page 69), cette comparaison montre que l'histogramme de couleurs fournit des résultats meilleurs que les blobs. Plusieurs explications ont été données pour cette dégradation de performance de blobs : (1) les blobs canoniques ne sont pas suffisamment descriptifs pour distinguer entre les catégories d'images et ils peuvent être mal fabriqués par le moyen du modèle gaussien diagonal; (2) l'apprentissage de l'arbre de décision prend en compte les blobs non pertinents ce qui dégrade les résultats; et (3) les 14 catégories d'images se chevauchent entre elles, en terme des régions, ce qui cause les difficultés.

Le travail récent effectué par Cadez et al. [17] sur la détection de l'anémia est le plus proche de notre approche en termes de données intermédiaires à classifier : les paramètres des mélanges gaussiens. Cadez et al. classifient les patients en deux catégories : patient normal et patient avec un manque de fer. Dans un premier temps, les données de bas niveau (40000 cellules sanguines) extraites de chaque patient sont modélisées par un mélange gaussien. Dans un second temps, la distribution des paramètres gaussiens estimées auparavant est de nouveau modélisée par un mélange gaussien de deux composantes (deux classes de patients). Cette approche est nommée hiérarchique car le mélange gaussien est appliqué deux fois. La deuxième fois, le mélange gaussien est utilisé pour la discrimination entre les deux types de patients. Des bons résultats sont obtenus par le mélange gaussien comparé avec d'autres classificateurs.

Cet aspect des données à plusieurs niveaux est aussi présent dans notre démarche. Cependant nos données posent quelques problèmes techniques : la taille de l'échantillon c'est-à-dire le nombre d'apparences d'un objet suivi est limité, les mélanges gaussiens des différents objets suivis n'ont pas des structures similaires (nombre de composantes par exemple) et le nombre de classes d'objets n'est pas connu a priori.

5.3 Notre approche

Cette section reprend les deux questions posées dans l'introduction de ce chapitre, et répond pour chacune d'entre elles dans les différentes étapes de l'approche de classification détaillées dans la suite. Dans le paragraphe suivant on présente un rappel du problème de la variabilité des objets suivis ainsi que de la solution proposée pour ce problème, tout en mettant en clair les données exploitées par la technique de classification. Ensuite, nous détaillerons la deuxième étape de l'approche qui consiste à mettre en correspondance les objets suivis modélisés par des densités de mélanges gaussiens en calculant les distances de Kullback et de Bhattacharyya . En se basant sur la matrice de proximités ainsi obtenue, l'algorithme de classification ascendante hiérarchique sera appliqué. C'est dans cette dernière étape que la sélection interactive du nombre de classes sera effectuée.

5.3.1 Hiérarchie des données

Les expérimentations menées dans les chapitres précédents ont montré que souvent la distribution d'un objet suivi dans l'espace de descripteurs de bas niveau est multimodale, à cause de la grande variation de l'apparence intra-plan de l'objet suivi et du manque de robustesse et d'invariance des descripteurs existant à ces changements de l'image. La modélisation de cette distribution par un mélange gaussien, dont la structure est déterminée automatiquement en fonction du degré de variabilité de l'objet suivi, consiste à identifier les classes d'apparences intra-plan de cet objet.

Ceci crée une nouvelle représentation de l'objet suivi dans l'espace de paramètres (μ, Σ). Ici, (μ, Σ) désigne le centre et la matrice de variances d'une composante gaussienne du mélange. Dans cet espace la représentation de l'objet suivi est plus compacte que celle dans l'espace initial de descripteurs; seulement quelques points multidimensionnels décrivent l'objet suivi (le nombre de ces points est équivalent au nombre des classes d'apparences de l'objet suivi). En revanche, la dimension de l'espace (μ, Σ) est égale à $d \times (d+1)$, donc beaucoup plus grande que celui de l'espace de descripteurs $\mathbb{R}^d$. La figure 5.1 illustre la distribution de l'enfant suivi de la figure 3.2 dans l'espace de l'histogramme de couleur et dans l'espace de paramètres gaussiens (μ, σ).

Pour résumer cette étape, les objets suivis représentés initialement dans l'espace de descripteurs $\mathbb{R}^d$ sont modélisés et ensuite représentés dans le nouvel espace de paramètres gaussiens $\mathbb{R}^{d\times(d+1)}$. Du fait que les objets suivis ne sont pas représentés en général par des points uniques dans ce nouvel espace, plusieurs algorithmes de classification, le mélange gaussien par exemple, ne peuvent pas être appliqués directement sur ces données.

5.3.2 Distance entre les objets suivis

Comme nous venons de voir, les données représentant les objets suivis dans l'espace de paramètres ne vérifient pas l'hypothèse d'unicité de description des individus pour une stratégie de classification classique; notamment plusieurs vecteurs de descripteurs (les paramètres (μ, Σ)) sont associés à chaque individu (un objet suivi).

Cette section discute le calcul d'une distance entre ces données comme une solution du problème de construction de classe sous ces conditions. Une formalisation de la distance

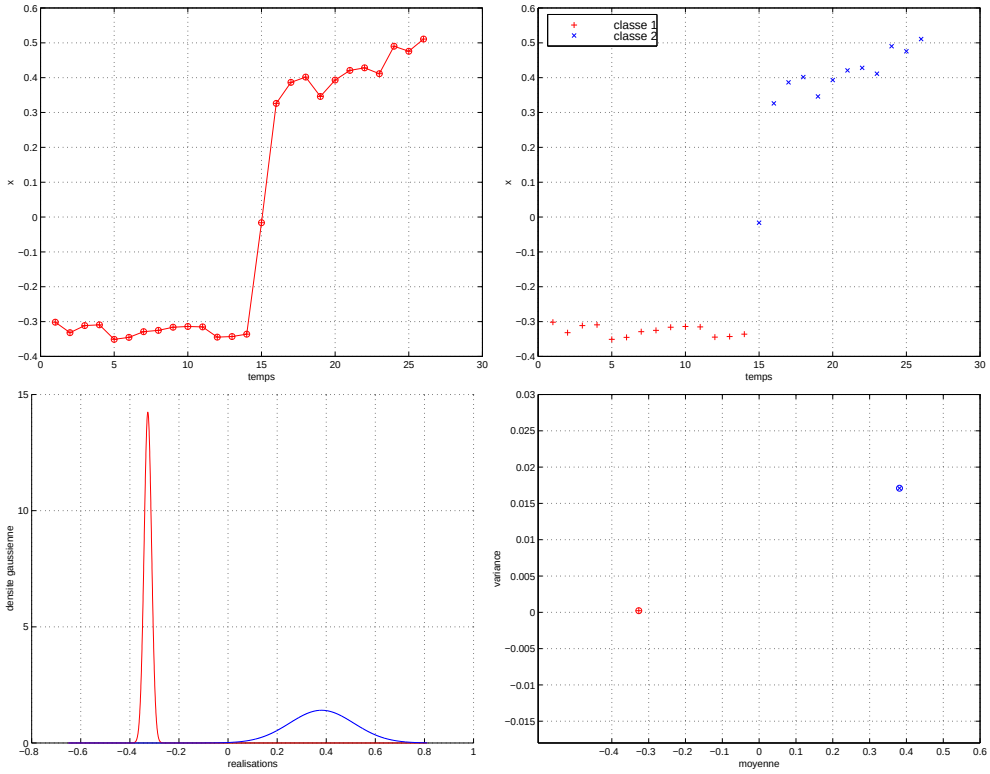


FIG. 5.1: *(a) Distribution de l'enfant suivi de la séquence Ajax (figure 3.2) : le premier axe factoriel de l'histogramme de couleur par rapport a l'axe du temps; (b) la partition correspondante décomposée en deux classes gaussiennes (obtenue par l'algorithme EM); (c) illustration de leurs densités gaussiennes; (d) nouvelle représentation de l'enfant suivi dans l'espace* (μ, σ).

entre deux objets suivis est décrite d'abord. Ensuite nous détaillons les deux distances de Kullback et Bhattacharyya .

5.3.2.1 Forme générale

Définition 5.3.1 *Nous considérons, une distance entre deux objets suivis,* $d_{\ell m}$*, comme étant le minimum des distances entre leurs classes d'apparences intra-plan,*

$$d_{\ell m} = \arg\min_{k,j}(\delta_{kj}) \tag{5.1}$$

avec $1 \leq k \leq K$ *et* $1 \leq j \leq J$*, K et J représentent le nombre de classes d'apparences intra-plan de deux objets suivis* ℓ *et m respectivement, et* δ_{kj} *représente la distance de Kullback ou de Bhattacharyya entre le couple* (k, j) *de classes d'apparences intra-plan. La formulation de ces distances est donnée dans le paragraphe 5.3.2.2.*

Justification de cette formulation Il est clair que deux objets suivis similaires ont très probablement des apparences intra-plan assez proches. Du fait que les apparences intra-plan de chaque objet suivi sont regroupées dans des classes homogènes et qu'une loi de mélange gaussiens les réunit, deux façons pour mettre en correspondance deux objets suivis peuvent être considérées :

1. *Appariement global* : de deux objets suivis en calculant une distance globale entre leurs densités de mélanges gaussiens.

2. *Appariement local* : de deux objets suivis en calculant des distances locales entre leurs composantes gaussiennes i.e. leurs classes d'apparences intra-plan.

Belongie et al. [7] choisissent la première voie pour apparier une image requête avec une image de la base; pour ce faire, ils utilisent dans leur système d'indexation d'images fixes la distance de Kullback calculée entre les densités de mélanges gaussiens. C'est dans l'espace $\mathbb{R}^5$, où les pixels d'une image sont représentés par leurs coordonnées spatiales et leurs trois canaux de couleurs RGB, qu'ils cherchent à partitionner l'image en régions homogènes par l'application de l'algorithme EM. L'explication que nous pourrions donner à cette décision est que deux images différentes peuvent avoir plusieurs (petites) régions semblables, et donc un calcul global de la distance peut empêcher une telle confusion et bien expliquer la divergence entre les deux images appariées. En revanche, une composante gaussienne de nos données représente une apparence intra-plan d'un objet suivi. Donc, le fait d'avoir deux classes d'apparences proches est une condition nécessaire et suffisante pour considérer que les deux objets suivis correspondant sont aussi similaires. Il suffit de l'identification de deux apparences pour conclure, et c'est le premier argument pour une similarité basée sur cette correspondance partielle des lois de mélanges.

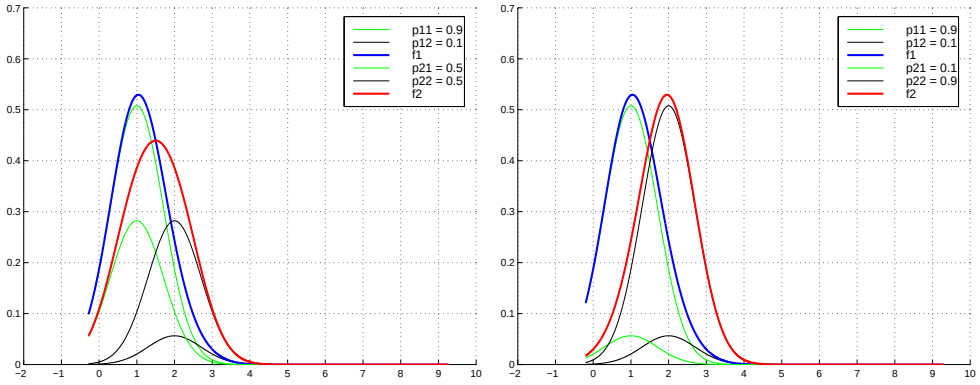


FIG. 5.2: *Illustration de l'effet de la variation du paramètre de proportion des composantes gaussiennes sur la forme générale de la densité de mélange. La densité de référence a pour paramètres* $(p_{11}, p_{12}, \mu_{11}, \mu_{12}, \sigma_{11}, \sigma_{12}) = (0.9, 0.1, 1, 2, 0.5, 0.5)$. *La densité de mélange modifiée a les mêmes paramètres que la densité de référence sauf les proportions qui vaulent* $(0.5, 0.5)$ *dans le premier exemple (figure du gauche) et* $(0.1, 0.9)$ *dans le second exemple (figure du droite).*

D'autre part, une distance globale entre deux densités de mélanges exploite tous les paramètres estimés de ces densités, c'est-à-dire les proportions, les moyennes et les matrices de variances des classes gaussiennes. Le paramètre de proportion représente le nombre des individus appartenant à une classe, et il est considéré comme étant la probabilité a priori d'une classe parmi la composition de la densité du mélange. Donc, ce paramètre a un effet important sur la forme – ou l'allure – de la densité de mélange. Par exemple, la figure 5.2 illustre l'effet de la variation de ce paramètre sur la forme de la densité du mélange gaussien, dans l'espace $\mathbb{R}^2$. Sur chaque figure, deux densités de mélanges, f_1 et f_2, chacune de deux composantes gaussiennes sont affichées. Les centres et les variances des composantes correspondantes sont égaux et seul les paramètres de proportions p_{ij} sont variés de telle manière que: $p_{11} + p_{12} = p_{21} + p_{22} = 1$. Cette petite expérience, montre clairement l'effet de la variation de la proportion, qui a causé une sorte de changement d'échelle entre les densités de chaque composante gaussienne et ensuite une translation de la densité du mélange. Ceci implique, une augmentation de l'écart (ou bien diminue l'intersection) entre les deux densités de mélanges. Sur cet exemple simple, le calcul d'une distance globale entre les densités f_1 et f_2 nous emmène à une décision que les deux populations (deux densités de mélanges) ne sont pas semblables; cette fausse conclusion n'est pas bonne car les deux populations ont des centres et des matrices de variances identiques. La figure 5.3 illustre la sensibilité de la distance de Kullback globale calculée entre les densités de mélanges gaussiens de l'exemple précédent (figure 5.2) à la variation du paramètre de proportion.

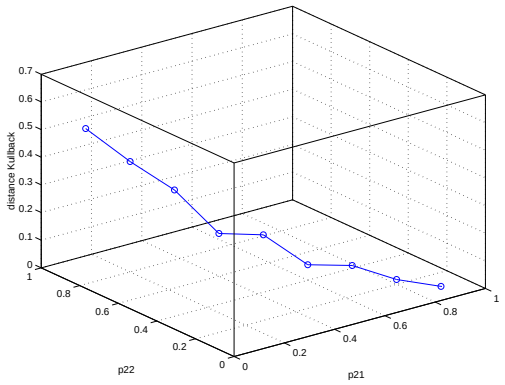


FIG. 5.3: *Illustration sur les données de l'exemple précédent de la sensibilité de la distance de Kullback globale à la variation du paramètre de proportion des composantes gaussiennes. La distance de Kullback est calculée entre la densité référence et celle dérivée de cette référence avec un changement des proportions seulement. Par exemple, entre les deux densités de mélanges gaussiens de la figure 5.2 (partie droite) la distance de Kullback mesure une divergence égale à* 0.4997 *qui est significative.*

En revanche, le calcul local d'une distance entre les composantes gaussiennes ne prend pas en compte le paramètre de la proportion i.e. le changement d'échelle entre les gaussiennes. Ceci est le deuxième argument pour lequel nous calculons une distance locale entre les composantes gaussiennes de deux objets suivis. En adoptant cette technique, les deux

populations de l'exemple précédent, où la distance de Kullback vaut zéro tout au long de la variation des proportions, sont jugées entièrement similaires. Ceci est très différent du contexte du travail de Belongie et al. [7] le paramètre de proportion est important à prendre en compte, par exemple, lorsqu'il s'agit de distinguer entre une petite et une grande tache rouge dans deux images.

5.3.2.2 Appariement des classes d'apparences intra-plan

Nous avons défini auparavant la forme générale de la distance entre deux objets suivis (équation 5.1), qui est exprimée en fonction de distance locale entre les classes d'apparences intra-plan. La mise en correspondance entre deux classes d'apparences est un cas particulier d'un problème général de comparaison de deux échantillons, où le but est de mesurer la similarité et pas pour décider si les échantillons sont identiques ou différents. Dans notre cas nous disposons de deux distributions gaussiennes multivariées, f_r et f_q, dont leurs paramètres $\theta_r = (\mu_r, \Sigma_r)$ et $\theta_q = (\mu_q, \Sigma_q)$ sont connus. Dans la suite, nous rappelons du principe général des tests d'hypothèses utilisés souvent pour comparer deux distributions et ensuite nous présentons quelques distances adaptées à nos données.

Tests d'hypothèses. Les tests d'hypothèses sont souvent utilisés dans la comparaisons de deux ou plusieurs distributions. Un test d'hypothèse comporte trois étapes :

- définition du test, par exemple,

$$\begin{cases} H_0\text{: les deux distributions sont identiques } \textit{hypothèse nulle} \\ H_1\text{: les deux distributions sont différentes;} \end{cases}$$

- résolution du test c'est-à-dire la construction de la fonction de décision permettant d'associer à toute réalisation de l'échantillon $(X_1, \ldots, X_n)$ soit l'hypothèse H_0, soit l'hypothèse H_1;
- et enfin la décision à prendre à partir d'une réalisation de l'échantillon.

La détermination de la fonction de décision revient à définir la partition de l'espace des valeurs de l'échantillon en deux sous-ensembles :

- l'ensemble W des réalisations pour lequelles on rejette H_0 en faveur de H_1 : c'est la *région critique*;
- l'ensemble $\bar{W}$ des réalisations pour lesquelles on conserve H_0 : c'est la *région d'acceptation.*

Soient α la probabilité de rejeter H_0 en faveur de H_1 alors que H_0 est vraie, et β la probabilité de conserver H_0 alors que H_1 est vraie (erreur de première et de deuxième espèce). Une illustration graphique de ces deux erreurs est donnée dans la figure 5.4. Souvent, c'est le risque de première espèce considéré comme le plus important par Neyman et Pearson, qu'on cherche à minimiser soit donc à maximiser la probabilité $1 - \beta$, appelée

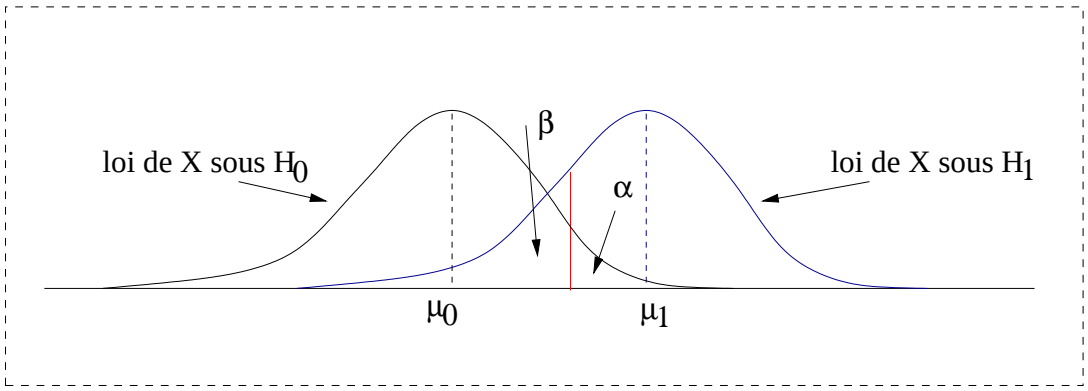


FIG. 5.4: *Les erreurs de première et deuxième espèce.*

puissance de test. La détermination de la fonction de décision revient alors à déterminer un sous espace W ayant, sous l'hypothèse H_0, une probabilité inférieure ou égale à $\alpha^\star$, qui est une valeur fixée a priori (souvent 5% ou 1%), appelée *niveau de signification du test*:

$$Prob[(X_1, \ldots, X_n) \in W \mid H_0] \leq \alpha^\star$$

Distance de Kullback Basée sur la théorie de l'information et des tests d'hypothèses, la distance de Kullback ([74] pages 3-6) est une distance bien adaptée pour la mesure de la divergence entre deux distributions quelconques, et en particulier entre deux distributions gaussiennes. Soient X une variable aléatoire dont la densité de probabilité dépend du paramètre θ, et les deux hypothèses H_0 et H_1 portant sur cette variable X, au vu d'une réalisation d'un échantillon $X_1, \ldots, X_n$, qu'elle provienne de la distribution gaussienne f_ℓ et f_m respectivement :

$$\begin{cases} H_0 \colon \theta = \theta_r \\ H_1 \colon \theta = \theta_q \end{cases}$$

En se basant sur le test paramètrique ci-dessus, la distance de Kullback est vue comme étant une fonction de décision permetttant d'associer à toute réalisation x de l'échantillon $X_1, \ldots, X_n$ soit l'hypothèse H_0 ou H_1. La divergence, $\delta(f_r, f_q)$, est ainsi définie par la somme de la moyenne des réalisations de θ_r qui conserve H_0 en faveur de H_1, notée par $I(0:1)$ et inversement.

$$\begin{aligned} \delta_k(f_r, f_q) &= I(0:1) + I(1:0) \\ &= \int f_r(x) log \frac{f_r(x)}{f_q(x)} d(\theta_r) + \int f_q(x) log \frac{f_q(x)}{f_r(x)} d(\theta_q) \\ &= \int log \frac{P(H_0|x)}{P(H_1|x)} d_{\theta_r}(x) - \int log \frac{P(H_0|x)}{P(H_1|x)} d_{\theta_q}(x) \end{aligned} \tag{5.2}$$

avec $P(H_i \mid x)$ est la probabilité conditionnelle d'avoir l'hypothèse H_i $(i = 1, 2)$ sachant la réalisation $X = x$.

L'évaluation de cette distance est réalisée par les procédures de Monte-Carlo qui consistent à générer des données simulées à partir des paramètres θ_0 et θ_1 et sur lesquelles

on estime les probabilités a posteriori ($P(H_i \mid x)$). La distance de Kullback présentée sous la forme ci-dessus est égale à zéro lorsque $\theta_0 = \theta_1$, elle est aussi symétrique, mais par contre, elle ne vérifie pas la propriété triangulaire d'une métrique.

Distance de Bhattacharyya . Un calcul direct d'une distance entre deux distributions gaussiennes, portant seulement sur les paramètres gaussiens (μ, Σ), est réalisable par le moyen de la distance de Bhattacharyya ([38], page 99) :

$$\delta_b(f_r, f_q) = \underbrace{\frac{1}{8}(\mu_r - \mu_q)^T \left(\frac{\Sigma_r + \Sigma_q}{2}\right)^{-1} (\mu_r - \mu_q)}_{B1} + \underbrace{\frac{1}{2} log \left(\frac{\mid \frac{\Sigma_r + \Sigma_q}{2} \mid}{\sqrt{\mid \Sigma_r \mid\mid \Sigma_q \mid}}\right)}_{B2} \tag{5.3}$$

où | M | dénote le déterminant de la matrice M. Le premier terme dans cette expression, $B1$, est similaire à la distance de Mahalanobis et il mesure la distance entre deux populations causée par le décalage de la moyenne, ainsi que le deuxième terme $B2$ exprime la séparabilité entre les classes grâce à la différence entre les covariances.

5.3.3 Classification hiérarchique

Nous avons abordé dans la section précédente le calcul d'une distance entre les objets suivis dans l'espace de paramètres gaussiens. En se basant sur la matrice de distances $\mathcal{D} = (d_{\ell m})_{1 \leq \ell, m \leq n}$ ainsi obtenue entre les n objets suivis, la troisième étape de notre approche consiste à appliquer la classification ascendante hiérarchique pour regrouper les objets dans des groupes homogènes.

Une hiérarchie consiste en une suite de partitions emboîtées, depuis l'ensemble de tous les n objets $\{1, \ldots, n\}$ jusqu'aux singletons formés par les objets eux mêmes, $\{1\}, \ldots, \{n\}$, en passant par des divisions successives des sous-ensembles. La procédure la plus couramment utilisée pour produire automatiquement une hiérarchie est la classification ascendante hiérarchique (CAH). Celle-ci se base sur une matrice de dissimilarité (ou de similarité) entre les objets, symétrique et de taille $n \times n$.

5.3.3.1 Algorithme du CAH

Les étapes d'un processus de classification ascendante hiérarchique sont les suivantes [69] :

1. Initialisation : chaque objet constitue une classe, on commence donc avec n classes qui sont des singletons. Les dissimilarités entre les groupes sont au départ les dissimilarités entre les objets qui les constituent.

2. Trouver les deux classes les plus similaires et les fusionner en une classe – on se retrouve ainsi avec une classe de moins.

3. Calculer les dissimilarités entre la nouvelle classe et les autres classes.

4. Répéter les étapes 2 et 3 jusqu'à ce que tous les objets soient réunis dans une classe de taille n, ou bien jusqu'à ce que l'on ait obtenu le nombre de classes désiré.

Le calcul de la dissimilarité entre deux classes à l'étape 3 peut être réalisé de différentes façons. Par exemple, si l'on utilise le critère d'agrégation du lien ou saut minimum (single-link clustering en anglais), la dissimilarité entre deux classes est définie comme la plus petite dissimilarité entre les objets des deux classes. D'autres critères d'agrégation peuvent être utilisés, comme celui du lien ou saut maximum ou celui de la distance moyenne (complete-link et average-link clustering), ou bien le critère d'inertie de Ward [124] (*Ward's minimum variance clustering*). Cependant, le critère de Ward est souvent utilisé dans la cas où les observations sont des vecteurs de $\mathbb{R}^d$ et les similarités des distances euclidiennes [107], ce qui ne correspond pas à notre cas.

5.3.3.2 Motivation du choix de CAH

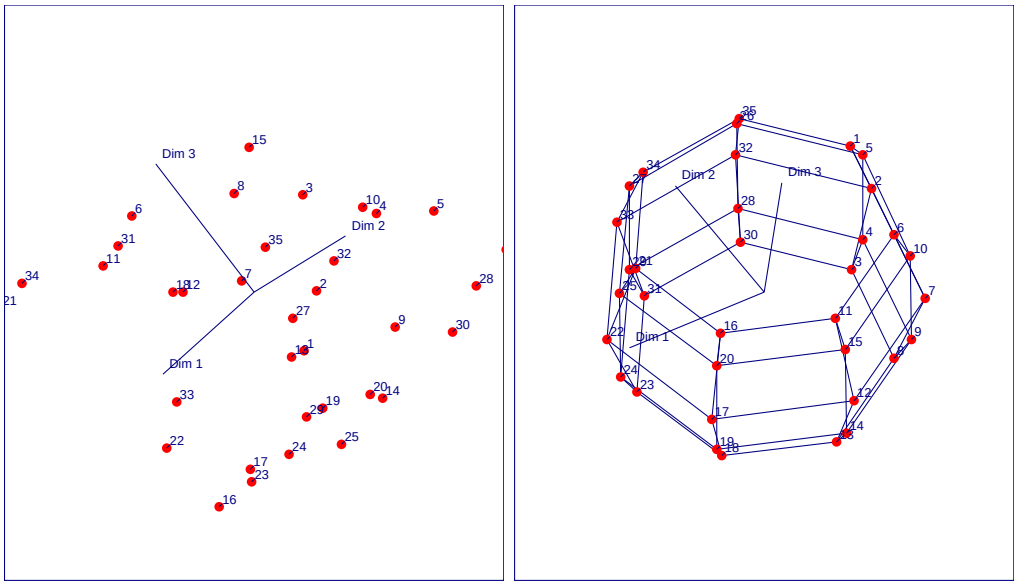
Dans le domaine de classification automatique de données, les méthodes de classification existantes (k-means, KNN, mélange gaussien, etc.), probabilistes et non probabilistes, peuvent être regroupées en deux familles – hiérarchique et partition – selon la structure de classes produite. Par exemple, le mélange gaussien est une méthode de classification probabiliste qui produit une partition unique des données, tandis que le CAH est hiérarchique et non probabiliste.

Nous avons employé la méthode de CAH pour les deux points suivants :

– *Nature des données exploitées :* Les données représentant les objets suivis dans l'espace de paramètres gaussiens ont une nature non habituelle pour les différentes méthodes de classifications, où chaque objets est représenté par plusieurs individus dans cet espace (voir section 5.3.1). Donc, on se retrouve avec une matrice de distances (de Kullback ou de Bhattacharyya) qui décrit la dissimilarités entre les objets suivis. Dans cette situation, la méthode de CAH est la seule applicable pour classifier automatiquement les objets. En revanche, d'autres méthodes de classification peuvent être appliquées sur la représentation spatiale de la matrice de distances fournie par exemple par la méthode de MDS [73]. Ainsi, l'objectif du MDS est de trouver pour les n objets suivis, notés par $x_1, \ldots, x_n$ et ayant comme matrice de distances $\mathcal{D}$, n points correspondants $y_1, \ldots, y_n$ de l'espace $\mathbb{R}^k$ (k petit, par exemple 2), avec une matrice de distances $\mathcal{D}_k$ qui soit la plus proche possible de $\mathcal{D}$ (voir figure 5.5). Il s'agit de minimiser la fonction de coût suivante :

$$\| \mathcal{D}(x_1, \ldots, x_n) - \mathcal{D}_k(y_1, \ldots, y_n) \|$$

Deux points nous ont amenés à renoncer à cette méthode non-linéaire : (1) la difficulté de trouver une représentation spatiale qui décrit parfaitement les données initiales, dans un espace de dimension réduite, et (2) la complexité en terme de calcul de la méthode de MDS pour trouver une telle représentation.



(a) (b)

FIG. 5.5: *Représentation spatiale obtenue par la méthode du MDS (b) pour la matrice de distances euclidiennes calculées entre les individus du nuage de points initial (a).*

– *Interprétation interactive des résultats :* Un des intérêt des méthodes hiérarchiques est qu'elles permettent d'obtenir une classification à différents niveaux de détail (voir figure 5.6) : plus on descend dans la hiérarchie, plus la classification est fine. Au contraire des méthodes qui produisent des partitions, la structure obtenue par le CAH permet une interprétation relativement intuitive du résultat, et l'auteur de la vidéo interactive bascule entre les niveaux de la hiérarchie sans aucun coût de calcul ou d'estimation de la partition désirée.

5.3.3.3 Quelques problèmes ouverts

Le choix du critère d'agrégation (saut minimum, maximum, etc.) reste un problème délicat. Théoriquement, tous ces critères donnent la plupart du temps les mêmes résultats si les classes sont compactes et bien séparées. Par contre, si les classes sont trop proches, ou n'ont pas une forme hypersphérique, des résultats très différents peuvent être obtenus. De plus les méthodes de classification hiérarchique n'optimisent généralement pas de critère numérique explicite, ce qui rend difficile le choix entre différentes méthodes et leur évaluation par rapport à un modèle de données par exemple. Néanmoins, il existe quelques exceptions; par exemple, la classification hiérarchique ascendante par la stratégie du saut minimum trouve, parmi les ultramétriques inférieures à la dissimilarité de départ, celle qui lui ressemble le plus. Ceci revient à trouver l'arbre de longueur minimum reliant tous les objets.

Le deuxième problème du CAH est le choix automatique du nombre de classes (niveau

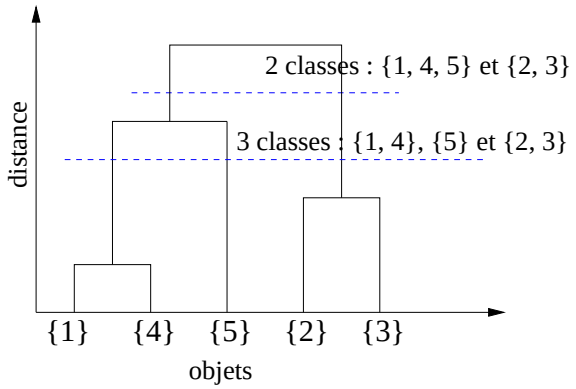


FIG. 5.6: *Exemple de hiérarchie : un dendrogramme correspondant à une classification hiérarchique de* 5 *classes initiales. Les traits pointillés montrent deux niveaux de détail que l'on peut choisir pour la classification finale.*

de coupure de la hiérarchie), où la théorie ne fournit pas des méthodes satisfaisantes. Nous abordons ce problème un peu plus en détail dans la section suivante.

L'algorithme général consiste à balayer à chaque étape un tableau de $\frac{n(n-1)}{2}$ distances afin d'en rechercher l'élément de valeur minimale, à réunir les deux objets correspondants, à mettre à jour les distances après cette réunion et à recommencer avec $n-1$ objets au lieu de n. La complexité d'un tel algorithme est de n^3 (ordre du nombre d'opérations à effectuer), qui est assez élevé. Cependant, diverses techniques ont été proposées pour accélérer les opérations [79]. Le lecteur peut s'adresser à la thèse de [99] pour plus de détails sur ce sujet qui sort du cadre de travail.

5.3.3.4 Sélection interactive du nombre de classes

Dans le domaine de la classification automatique ou non supervisée de données le nombre de classes est inconnu, et on ne sait pas de quelles classes proviennent les objets à classifier. Le fait d'avoir un critère automatique pour sélectionner le nombre de classes est très important, en particulier lorsqu'il s'agit d'un travail répétitif de classification de données. C'est par exemple le cas où nous avons cherché à classifier les apparences intra-plan de chaque objet suivis dans la vidéo par un mélange gaussien. Il est clair dans ce cas qu'un nombre fixe de classes d'apparences pour tous les objets suivis n'est pas le bon choix, vu que ces objets n'ont pas le même degré de variabilité.

Il existe quelques méthodes pour choisir le point de coupure de la hiérarchie formée par le CAH [66] [84]. Une étude comparative sur des données synthétiques a été menée par Milligan et Cooper [84]. Le critère le plus connu est basé sur la minimisation de la variance intra-classe et la maximisation de la variance inter-classe [97]. Il est connu que l'utilisation de la variance sur ce type de problème favorise la formation de classes hyper-sphériques. Une analyse de données ne vérifiant pas cette hypothèse implicite est donc biaisée. Sur l'exemple de la figure 5.7, donnée dans [100], la plupart de ces méthode détecterons vraisemblablement la présence de quatre classes dans les données. Les trois

classes les plus denses sont fusionnées en une seule. Une méthode récente a été proposée par Ribert [100] qui consiste à couper la hiérarchie à plusieurs échelles, en prenant en compte le cas des classes de densités différentes. Cette méthode consiste à estimer un critère σ^2/μ^2 qui se fait sur les valeurs de l'histogramme calculé sur une hiérarchie. Par exemple, il calcule un histogramme à N intervalles sur une hiérarchie donnée. La variance et la moyenne seront estimées sur les N valeurs de l'histogramme (et non pas sur les valeurs des palliers de la hierarchie). Pour obtenir les valeurs critiques du critère, il faut établir un abaque f(*dimension de l'espace, nombre d'éléments*). Le principe est de générer des configurations avec la distribution attendue (ex. gaussienne) ne comprenant qu'une seule classe. Dans le cas où la dimension de l'espace est connue, il faut générer des bases synthétiques dans un espace de même dimension. Il est en effet important de prendre en compte ce paramètre car l'interprétation d'une hiérarchie peut changer en fonction de la dimension de l'espace de représentation. Cette méthode permet de résoudre quelques cas de densité variable (voir figure 5.7), où les méthodes de coupure unique (horizontale) échouent. L'inconvénient majeur (d'après l'auteur de cette méthode) réside dans la constitution d'une abaque de référence.

Vue la grande variation de l'apparence inter-plan des objets suivis à classifier, nous sommes quasiment convaincus que tous ces critères ne sélectionnerons pas le meilleur niveau de coupure de la hiérarchie.

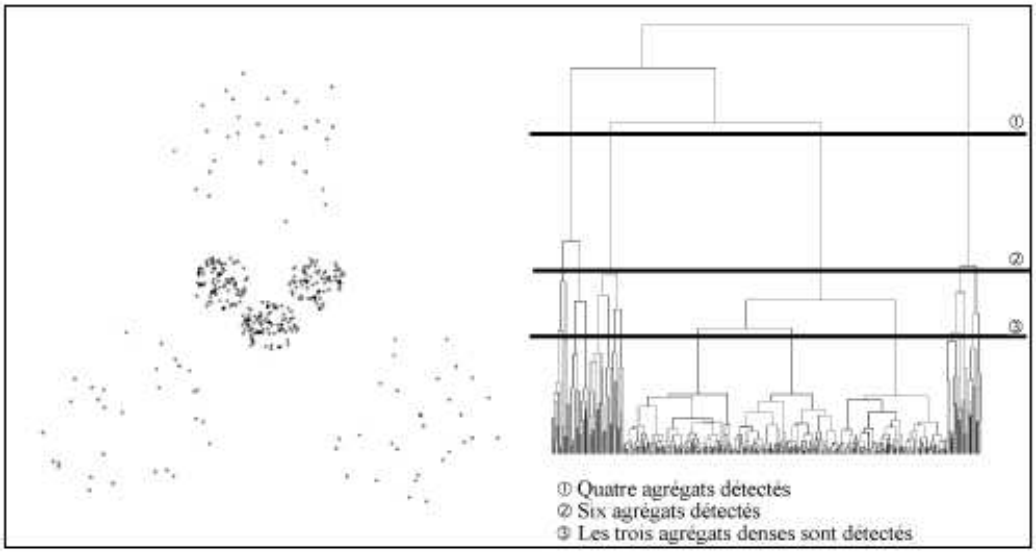


FIG. 5.7: *Exemple de densité variable de* 6 *agrégats (classes); échec de la coupure unique de la hiérarchie.*

Nous proposons à l'utilisateur expert une méthode interactive et visuelle pour sélectionner le nombre de classes des objets suivis. D'abord on filtre les niveaux de la hiérarchie fabriquée qui ont des distances de transition (distance relative) assez importante (par rapport à un seuil pré-défini), ensuite l'utilisateur expert choisit un niveau parmi eux, il

visualise les classes obtenues à ce niveau, et enfin il fixe son choix pour un de ces niveaux.

Comme on ne peut pas s'attendre à un résultat parfait le système pour la construction de la hyper-vidéo fournit à l'utilisateur expert des outils interactifs qui lui permettent de corriger les résultats de la classification. Ces outils sont principalement du type "drag & drop" et permettent de déplacer par la souris un objet mal classé d'une classe à une autre déjà existant ou non. Plus de détails sur cet aspect interactif dans le chapitre 8.

5.4 Expérimentation

5.4.1 La séquence Avengers-2

Les expérimentations de l'approche proposée ont été menées sur la séquence Avengers-2 du film "Chapeau Melon et Bottes de Cuir". En appliquant les outils de segmentation de la vidéo présentés au chapitre 2, 2749 apparences d'objets correspondant à 29 objets suivis ont été localisées dans 1938 images. Ces apparences d'objets correspondent à 7 classes d'objets différentes (classes de *J. Steed, Licorne, voiture Ford-I, voiture Ford-II, Purdey, Prêtre* et *voiture Mercedes*). La figure 5.8 illustre quelques objets de cette séquence. Comme cette figure le montre, les objets suivis de Avengers-2 sont mobiles et d'apparences intra- et inter-plans très variables. A la différence avec la séquence Avengers-1 chaque objet suivi de Avengers-2 possède une trentaine d'apparences intra-plan au moins. Ce nombre d'apparences intra-plan par objet est relativement suffisant vis-à-vis de la dimension réduite de l'espace de descripteurs. Cette situation des données permet de tester l'approche proposée dans les meilleurs conditions d'estimation et de comparaison des paramètres gaussiens. Les classes d'apparences intra-plan des différents objets suivis seront modélisées par le même type de modèle de mélange gaussien afin qu'on soit capable de les comparer.

5.4.2 Paramètrage de l'approche

La modélisation de la variabilité intra-plan des objets suivis est une étape fondamentale de nos approches de classification. Les deux chapitres précédents ont évoqué cette étape du point de vue théorique et expérimentale. En effet, des restrictions sont posées sur les expérimentations présentées ici en vu des résultats obtenus auparavant : *meilleur espace de descripteurs* et *structure optimale du mélange gaussien.*

Descripteurs de bas niveaux L'adoption des descripteurs de couleurs dans nos expérimentations est un choix déjà discuté dans la section 3.3. Ici, nous calculons sur les apparences des objets suivis les histogrammes RGB, HSV, I, et H. Ces histogrammes sont quantifiés d'abord en 64, 64, 32 et 32 cellules respectivement. Ensuite, une analyse en composante principale est appliquée sur chaque nuage de descripteurs.

Données de classification Comme nous l'avons mentionné auparavant, les données sur lesquelles la classification hiérarchique sera appliquée sont les paramètres gaussiens des classes d'apparences intra-plan des 29 objets suivis de Avengers-2. Le nombre maximal de composantes gaussiennes d'une loi de mélange est fixé à 3 ($MaxNbC = 3$), et seul

FIG. 5.8: *Quelques apparences d'objets suivis de la séquence vidéo Avengers-2.*

le modèle gaussien général ($[\lambda_j Cj]$) est employé. D'une part ce modèle gaussien estime la frontière optimale entre les classes d'apparences intra-plan d'un objet suivi, et d'autre part le risque d'estimation instable est très faible vu le nombre suffisant d'apparences d'objets suivis de la séquence Avengers-2 par rapport à la dimension réduite des espaces de descripteurs (10 pour RGB et HSV, et 5 pour I et H). Aussi, fixer un modèle gaussien est logique car le calcul d'une distance de Bhattacharyya par exemple entre deux modèles gaussiens de différent types n'a pas de sens : par exemple un modèle gaussien multivarié de la famille sphérique (variables indépendantes) et un second de la famille générale (variables dépendantes) ne sont pas comparables (voir section 3.6.2.3). Dans ce cas le critère ICL servira seulement à choisir le nombre de composantes gaussiennes le mieux adapté aux données.

Dans l'évaluation de la distance de Kullback par la procédure de Monte Carlo nous générons environ 2000 individus simulés à partir de chaque composante gaussienne référence. Ce nombre est choisi empiriquement mais par contre il est beaucoup plus élevé que la taille de l'échantillon utilisé dans l'estimation des paramètres gaussiens, ce qui veut dire que l'échantillon simulé conduit à une évaluation de la distance stable. Notons que les valeurs de cette distance varient en fonction de la taille des données simulées. En terme de complexité les deux distances implantées se calculent en temps réel sur une machine UltraSparc 256 MHZ mais avec un ordre de grandeur différent; le calcul de la distance de Bhattacharyya est 10 fois plus rapide que celui de la distance de Kullback , car elle est formulée directement sur les descripteurs et ne possède pas une phase de simulation de données.

Maintenant, si les distances calculées ont des valeurs petites entre les objets de la même classe sémantique (distances intra-classes) et des valeurs grandes entre les objets des classes différentes (distance inter-classes) alors on peut s'attendre à une classification automatique parfaite. Sinon, il y aurait certainement des mauvaises classifications. Cela peut s'expliquer comme il suit. La distance est peut-être mal adaptée pour la comparaison des composantes gaussiennes. Ou on est dans le cas où des distances inter-classes sont plus petites que celles intra-classes. Ce dernier cas est dû très probablement à la grande variance dans l'apparence des objets de la même classe sémantique.

La figure 5.9 illustre la matrice de distance de Kullback obtenue durant cette expérimentation pour 9 objets suivis. Parmi ces 9 objets, $1, 2, 3, 4$ et 5 appartiennent à la classe sémantique "Steed" et le reste appartient à la classe "Espion". Sur cette représentation graphique on observe que les distances intra-classes ont des basses fréquences inversement aux distances inter-classes. Selon cette représentation des proximités on peut nettement classifier les objets en deux classes différentes qui correspondent parfaitement aux deux classes sémantiques.

5.4.3 Résultats et procédure d'évaluation

L'application de l'algorithme de CAH sur les matrices de distances calculées produit des hiérarchies de 29 niveaux de détail; la coupure horizontale de la hiérarchie à un certain niveau i génère $(29 - i + 1)$ groupes d'objets suivis.

A un certain niveau de la hiérarchie la qualité des résultats de regroupement est ex-

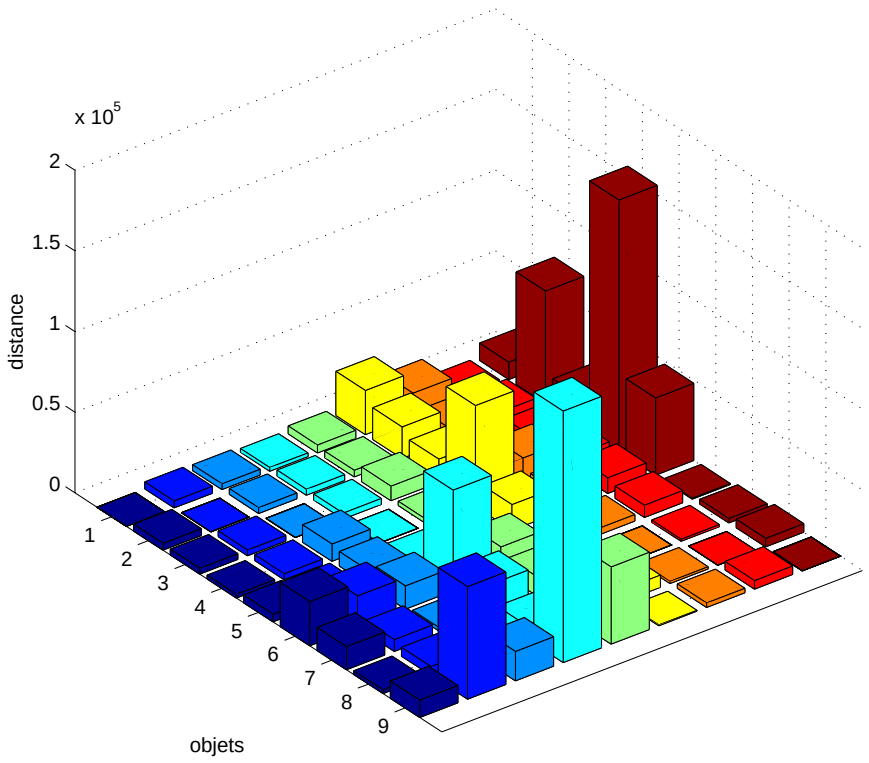


FIG. 5.9: *Illustration de la matrice de distances de Kullback entre* 9 *objets de la séquence Avengers-2; Les objets* $1, 2, 3, 4$ *et* 5 *appartiennent à la classe sémantique "Steed" et le reste appartient à la classe "Espion". Cette matrice de distances décrit bien cette partition.*

primée en fonction de l'homogénéité des individus de chaque classe. Les objets d'une classe sont jugés homogènes ou non selon la vision humaine.

Soit $cc\%$ le pourcentage de classification correcte pour K classes d'un niveau i. L'algorithme d'évaluation proposé ici est le suivant :

0- Initialiser le nombre des objets correctement classifiés dénoté par cc à 0;

1- Compter le nombre des éléments homogènes dans chaque classe k ($1 \leq k \leq K$). Par exemple, la classe numéro 1 regroupe 5 objets suivis, 4 objets de type-1, 1 objet de type-2 et 0 objets de type-$[3...L]$, avec L le nombre correct de classes d'objets fabriquées manuellement ($L = 7$ pour les objets de la séquence Avengers-2). Soit $S = (s_{k\ell})$ la matrice $K \times L$ des scores obtenues pour les K classes; l'élément $s_{k\ell}$ représente le nombre des objets de type-ℓ trouvés dans la classe k.

2- Pour $k = 1$ jusqu'à K faire

   a- calculer $A = \max(s_{k\ell})_{1 \leq \ell \leq L}$; soit j le numéro du colonne de l'élément A dans la matrice S

b- calculer $B = \max(s_{ij})_{k \leq i \leq K}$;

c- **Si** $A \geq B$ **alors** $cc = cc + A$;
affecter les éléments s_{ij} avec $(k+1) \leq i \leq K$ à zéro; (cette opération associe un label unique "type-ℓ" à la classe k);

d- **Sinon**, affecter l'élément $s_{k\ell}$ à zéro, et répéter l'étape a.

3- $cc\% = cc/n \times 100$, où n désigne le nombre total des objets classifiés.

Cet algorithme est utilisé pour évaluer les résultats obtenus au niveau 22 des hiérarchies des différents descripteurs de bas niveaux de cette expérimentation. Ce niveau est choisi car en réalité tous les objets suivis de la séquence Avengers-2 correspondent à 7 classes sémantiques différentes, comme nous l'avons déjà mentionné dans ce chapitre.

Le tableau 5.1 résume les résultats de test. Le meilleur pourcentage de classification correcte est de 79.31%. Les figures 5.10 et 5.11 illustrent les 7 classes d'objets trouvées automatiquement lorsque les paramètres gaussiens sont estimés dans l'espace de descripteurs RGB, la distance de Kullback est calculée et le critère de saut maximum est employé Ces résultats avec les séquences vidéo complètes sont disponible sur le Web à l'adresse suivante :

http://www.inrialpes.fr/movi/people/Hammoud/unsupervisedClassification.htm

Espace de descripteurs	d	d_E	Distance	$cc\%$ Critère du saut		
				minimum	maximum	moyenne
h_{RGB}	64	10	δ_b	44.83	62.07	62.07
h_{RGB}	64	10	δ_k	55.17	79.31	68.97
h_I	32	5	δ_b	44.83	65.52	58.62
h_I	32	5	δ_k	48.28	68.97	58.62
h_{HSV}	64	10	δ_b	44.83	65.52	68.97
h_{HSV}	64	10	δ_k	51.72	68.97	58.62
h_{Hue}	32	5	δ_b	55.17	55.17	58.62
h_{Hue}	32	5	δ_k	58.62	62.07	58.62

TAB. 5.1: *Les pourcentages de classification correcte des d'objets suivis de la séquence Avengers-2, dans le cas où la variabilité intra-plan des objets suivis est modélisée par des densités de mélanges gaussiens dont le nombre maximal de composantes permis est* 3.

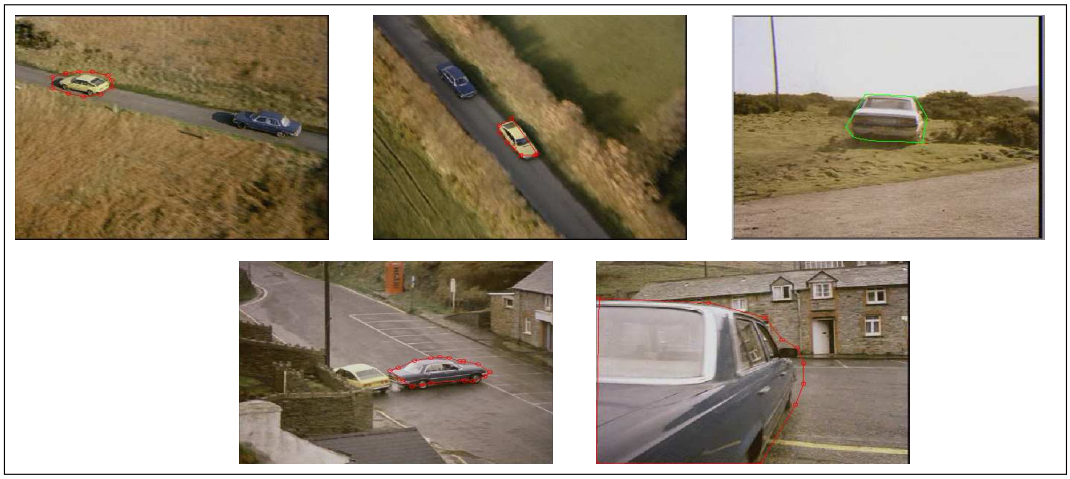
5.4.4 Analyse comparative et discussion

La classification automatique de données est une tâche difficile en général. De plus, lorsqu'il s'agit de classifier des densités de mélange gaussien qui représentent des objets suivis non-rigides et d'apparences trop variable dans les scènes, on n'attend pas une construction automatique des classes entièrement parfaite. Les tests sur la base réelle d'objets suivis

FIG. 5.10: *Les groupes d'objets suivis de la séquence Avengers-2 fabriquées automatiquement. La suite de ces résultats est illustrée par la figure 5.11. Une seule apparence par objet suivi est illustrée ici; les* 2745 *apparences de ces objets suivis sont disponibles sur le web à l'adresse mentionnée auparavant. Le pourcentage de classification correct obtenu est égal à* 79.31%.

"Classe Ford-I"

"Classe Mercedes"



"Classe Ford-II"

FIG. 5.11: *Suite de la figure 5.10*

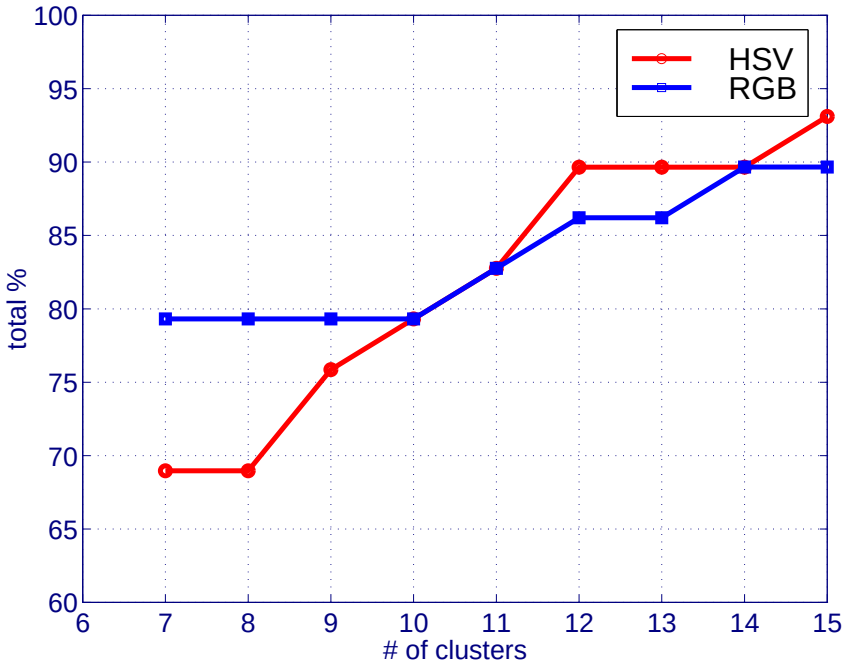


FIG. 5.12: *Illustration du pourcentage de bonne classification des objets suivis par rapport au nombre de classes dans la hiérarchie, dans le cas où le critère du saut maximum et la distance de Kullback sont employés, et les paramètres gaussiens sont estimés dans les espaces de descripteurs RGB et HSV.*

de la séquence vidéo Avengers-2 donnent un pourcentage de bonne classification de 80% environ; ce qui revient à une classification correcte de 23 objets suivis parmi les 29 de la séquence. On peut remarquer que les 6 objets mal classés sont partagés dans les deux dernières classes *Ford-I* et *Ford-II* souvent nommées classes poubelles en classification automatique. Une classe poubelle est le fait d'une coupure de la hiérarchie à une échelle unique; elle regroupe généralement des éléments non-homogènes forcés à être ensemble. Une coupure unique de la hiérarchie à un niveau plus bas que 22 ou bien une coupure à plusieurs échelles permet probablement de garder les 5 premières classes et de couper les deux dernières classes en quatre sous-classes plus homogènes. Dans ce cas, une surestimation du nombre de classes d'objets est effectuée mais par contre le contenu des classes est plus homogène. La figure 5.12 illustre le pourcentage de bonne classification par rapport aux nombres de classes. On peut voir que lorsque le nombre de classe est double du vrai nombre, un pourcentage de 95% de bonne classification est réalisé. Pratiquement, le pourcentage obtenu n'a pas atteint le 100%. Cela est expliqué par le fait que certains objets mal classifiés sont certainement regroupés au début de la classification hiérarchique ascendante. Malgré ces cas inévitables la construction des classes à des niveaux prédécesseurs de 22 améliore considérablement le pourcentage de classification correcte. La généralisation de cette technique "fournir à l'utilisateur le double du nombre de classes d'objets qui existent en réalité" permet de simplifier considérablement le coût d'interaction manuelle

effectuée par l'auteur de la vidéo hyperliée qui corrigera les résultats de classification. Il s'agit d'éditer par exemple $2m$ classes au lieu de $n \times m$ classes, si on suppose qu'il s'agit de classifier m objets suivis qui apparaissent n fois dans la séquence vidéo traitée.

Le choix interactif du nombre de classes ou du niveau de coupure de la hiérarchie se fait par l'auteur de la vidéo hyperliée, soit directement par précision du nombre désiré, soit par sélection d'un point de courbure sur la courbe des indices hiérarchiques (distance minimale entre groupes) aux itérations du CAH. La figure 5.13 illustre la courbe des indices hiérarchiques par rapport au nombre de classes, dans le cas où le critère du saut maximum est employé, et les paramètres sont gaussiens estimés dans les espaces de descripteurs RGB et HSV. En filtrant les points de courbures les plus significatifs pour ces courbes (par exemple 9, 7, etc), la tâche devient plus simple à l'utilisateur pour prendre sa décision. Dans le système développé pour ce travail, l'utilisateur pourra visualiser les classes d'objets à un niveau désiré de la hiérarchie, ensuite servir des outils de correction manuelle de résultats obtenus (voir section 5.3.3.4).

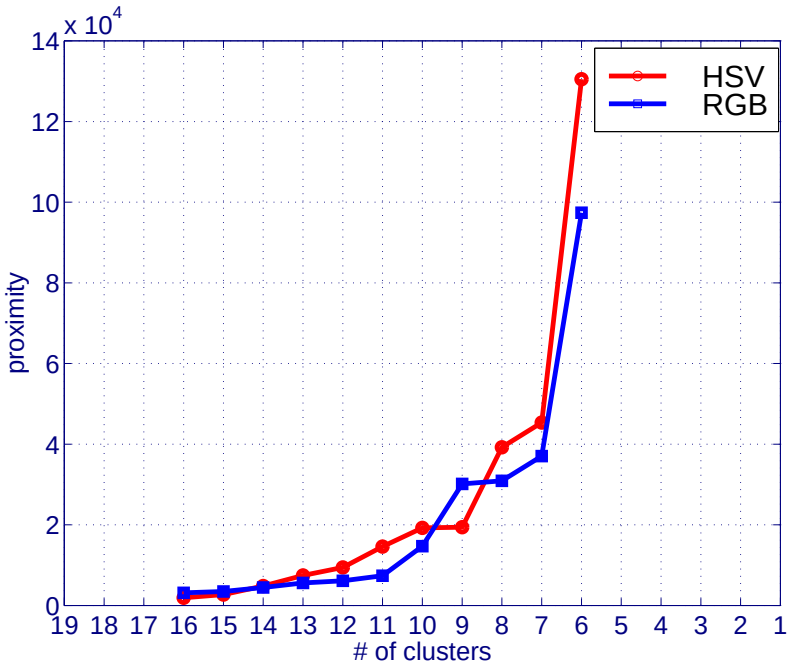


FIG. 5.13: *Illustration de l'indice de la hiérarchie (distance minimale entre groupes) par rapport au nombre de classes, dans le cas où le critère du saut maximum est employé, et les paramètres sont gaussiens estimés dans les espaces de descripteurs RGB et HSV.*

Comme nous avons mentionné dans la section 5.3.3.3, un problème du choix de la distance entre groupes se présente à l'application de la classification hiérarchique. Les expériences ci-dessus montrent que le critère de saut maximum fournit les meilleurs résultats. Par contre, les mauvais résultats sont obtenus quand le critère de saut minimum est employé. Un peu de chevauchement entre les groupes non homogènes à une itération i du CAH conduit à un fusionnement par le critère de saut minimum lors de la prochaine

itération. Par contre, le critère du saut maximum retarde ce fusionnement le plus possible. Le critère de la distance moyenne un est compromis entre les deux sauts minimum et maximum et devra fournir des résultats meilleurs que les deux, mais expérimentalement ceci n'est pas vérifié. Une augmentation des pourcentages de bonne classification allant de 10% jusqu'à 20% est obtenue par l'adoption du critère du saut maximum.

La distance entre deux objets suivis est considérée comme la distance minimale entre leurs classes d'apparences intra-plan. Deux distances de Kullback et de Bhattacharyya ont été calculées entre ces classes d'apparences; chaque classe d'apparences est modélisée par une gaussienne. Ainsi, les tests montrent que la distance de Kullback fournit toujours des résultats de classification meilleurs que la distance de Bhattacharyya , allant de 3% jusqu'à 17%.

Mieux modéliser la variabilité intra-plan des objets suivis induit une meilleure classification. Cette règle est aussi validée dans la classification automatique quand la modélisation est effectuée dans les espaces de descripteurs RGB et I. Le tableau 5.2 résume les résultats de la classification automatique lorsque la variabilité des objets suivis est modélisée par une seule gaussienne. La comparaison de ces résultats avec les précédents (tableau 5.1) prouve une amélioration de 10% à 16% quand le mélange gaussien est adapté dans les espaces RGB et I. Par contre, les résultats sont presque similaires pour les autres espaces de descripteurs. Cette dernière situation reste pour nous inexpliquées.

Espace de descripteurs	d	d_E	Distance	$cc\%$ Critère du saut		
				minimum	maximum	moyenne
h_{RGB}	64	10	d_B	44.83	55.17	55.17
h_{RGB}	64	10	d_K	44.83	62.07	62.07
h_I	32	5	d_B	48.28	55.17	55.17
h_I	32	5	d_K	41.38	55.17	51.72
h_{HSV}	64	10	d_B	51.72	62.07	65.52
h_{HSV}	64	10	d_K	51.72	62.07	62.07
h_{Hue}	32	5	d_B	51.72	58.02	62.07
h_{Hue}	32	5	d_K	55.17	62.07	62.07

TAB. 5.2: *Les pourcentages de classification correcte des d'objets suivis de la séquence Avengers-2, dans le cas où la distribution de chaque objet suivi est modélisée par une seule composante gaussienne.*

5.5 Conclusion

Dans ce chapitre nous avons décrit une approche de classification automatique des objets suivis. Le but d'une telle approche est d'aider au maximum l'auteur de la vidéo hyperliée à construire la structure de "groupes d'objets". Dans cette approche la classifica-

tion ascendante hiérarchique est employée pour fabriquer des partitions imbriquées. D'un point de vue technique le choix de cette méthode est motivé par le fait que les classes d'objets fabriquées peuvent être examinées par l'auteur de la vidéo hyperliée à différents niveaux sans aucun coût de calcul.

Ici, les données à classifier ne sont pas des descripteurs classiques mais des densités de mélanges gaussiens. Les classes d'apparences intra-plan de chaque objet suivi sont d'abord recherchées dans les espaces de descripteurs de couleurs. Ensuite, les distances de Kullback et de Bhattacharyya sont calculées localement.

Les expérimentations ont été menées sur la séquence vidéo Avengers-2 dont le contenu est très complexe en termes de variation de l'apparence intra- et inter-plan des objets suivis. Sur une telle base d'objets bruités et pour un processus de classification automatique les résultats obtenus sont vraiment intéressants. Un pourcentage de bonne classification de 80% est atteint. Ces résultats sont jugés acceptables pour une initialisation qui serait reprise par l'auteur de la vidéo hyperliée. Le choix du nombre de classes est effectué par l'utilisateur d'une manière interactive. Aussi, des outils interactifs sont fournis par le système développé pour ce travail afin de permettre une édition des résultats. Ces expérimentations permettent de suggérer de procéder ainsi:

- Modéliser la variabilité intra-plan des objets suivis par un mélange gaussien;
- Utiliser la distance de Kullback qui est mieux adaptée que celle de Bhattacharyya à la comparaison des densités gaussiennes;
- Employer la distance de "saut maximum" dans l'application de la classification ascendante hiérarchique.

Les données classifiées ici sont très bruitées. On ne peut donc pas s'attendre à une classification automatique parfaite. Une approche de "relevance feedback" adoptée lors de la construction de la hiérarchie des classes d'objets, surtout les premiers niveaux, permettrait vraisemblablement d'améliorer la qualité de la classification.

Chapitre 6

Extraction des apparences-clés

Je m'autoriserai des temps de pause.

Le niveau le plus élémentaire dans la structure hiérarchique d'un film vidéo, illustrée dans la figure 2.1, est celui des *images-clés*. Visuellement quelques images fixes choisies à différents instants de l'axe temporel, nommées "images-clés", peuvent décrire suffisamment le contenu (ou l'histoire) du plan vidéo. Lorsque des objets sont suivis dans les plans alors on introduit le niveau des *apparences-clés* .

Dans ce chapitre nous nous intéressons au problème de l'extraction automatique des apparences-clés d'un objet suivi. Notons qu'une relation forte existe entre apparence clé et image clé mais sans généralisation; par exemple le cas d'une scène avec un changement du fond (plusieurs images-clés) et sans variation des apparences des objets (une seule apparence clé).

La sélection des images-clés est un problème rencontré dans différents domaines d'applications: présentation, indexation, codage et transmission de la vidéo. Quelles sont les images les plus informatives à présenter à l'utilisateur de la vidéo structurée? afin de lui permettre une visualisation rapide du contenu de plan (les images-clés pointent sur les plans correspondants dans la vidéo, un clic sur une image clé permet par exemple de jouer le plan correspondant; idem pour la mosaïque d'objets [56]), et/ou une vue globale du contenu du film (balade dans la mosaïque de plans, voir figure 4.1 par exemple), tout en prenant en compte la capacité limitée des écrans d'ordinateurs et le nombre important de plans d'un film vidéo ($500 - 3000$ plans). En ce qui concerne l'indexation de la vidéo par le contenu, la similitude entre les plans d'une vidéo se fait souvent sur la base des images clés où l'image médiane (ou de début/fin) du plan est considérée comme image représentative [8] [92] [130] [53]. Cependant, ce choix simplificateur conduit à des taux de bons appariements des plans (ou objets) très faibles [57] pour les raisons que nous avons discutée dans le chapitre 4.

Un résumé de l'approche de sélection des images- apparences-clés que nous avons proposée est décrit dans la section suivante. Cette approche est expérimentée sur plusieurs

objets suivis du film "Avengers" et sa mise en oeuvre dans l'application de la vidéo interactive est seulement exploitée pour un objectif de présentation (*browsing*) intelligente de l'hyper vidéo.

6.1 Résumé de "A Probabilistic Framework of Selecting Effective Key-Frames for Video Browsing and Indexing" - RISA'2000

Cet article, publié à RISA'2000, concerne la séléction des apparences intra-plan clés des objets suivis par une méthode de classification non-supervisée. Les outils de segmentation en plan et de suivi d'objets sont présentés dans le chapitre 2. L'idée de base de cette approche consiste dans un premier temps à regrouper les apparences d'un objet suivi qui sont proches dans l'espace de descripteurs de bas niveau en des classes homogènes. Pour ce fait le mélange gaussien est employé comme une méthode de classification automatique. Cette technique ainsi que le traitement des données expérimentales ont été présentés en grand détail dans le chapitre 3. Dans un second temps, un algorithme adaptatif permet de sélectionner les apparences-clés des classes d'apparences retenues dans la première étape. D'abord l'apparence médiane de chaque classe d'apparences est sélectionnée. Ensuite une phase de test de la compacité des classes d'apparences à l'aide de la distance de Mahalanobis permet d'étendre le nombre des apparences-clés par classe, et une dernière phase de vérification temporelle consiste à garder le nombre minimal des apparences-clés de l'objet suivi et qui soient suffisamment séparés dans le temps. Une description plus détaillée de cet algorithme est présentée dans les pages suivantes.

A Probabilistic Framework of Selecting Effective Key-Frames for Video Browsing and Indexing

Riad Hammoud and Roger Mohr

International workshop on **R***eal-Time* **I***mage* **S***equence* **A***nalysis, pages 79-88*
August 2000

Abstract: To represent effectively the video content, for browsing, indexing and video skimming, the most characteristic frames (called key-frames) should be extracted from given shots. This paper, briefly reviews and evaluates the existing approaches of key-frames extraction; and then introduces a framework of selecting effective key-frames using an unsupervised clustering method. The mixture of Gaussians is used to model the temporal variation of the feature vectors of all frames in the shot. As a result, the feature-based representation of the shot is partitioned into several clusters. From each obtained cluster, firstly the closest frame to the median of its frames is selected as a reference key-frame. Then depending on the variation in time and appearance of the cluster content against the reference key-frame multiple frames can be extracted to represent effectively the cluster. The number of clusters is determined automatically by the Bayes Information Criterion. Experimental results on tracked objects in a real-world video stream are presented which illustrate the performance of the proposed technique.

1 Introduction and motivation

As the amount of video data grows rapidly, the ability to manipulate it efficiently becomes of greater importance, for the purpose of selection of appropriate elements of information [6]. The selection or extraction of limited and meaningful informations is a way to resolve a set of challenging problems for recently emerging multimedia applications: video browsing and navigation, content-based indexing, video summarization and trailers, storage and transmission bandwidth of digitized video information [14] [9].

The access to video is still a hard task due to video's length and unstructured format. Video abstraction and summarization techniques are needed to solve this difficulty. Shot boundary detection and key-frame extraction are two bases for abstraction and summarization techniques [15] [1] [11].

A *shot* is defined as an unbroken sequence of frames recorded from a single camera, which forms the building block of a video. The purpose of shot boundary detection is to segment the video stream into multiple shots. There exist many already effective shot boundary techniques [2].

Beyond the shot level an abstraction level could be constructed by mapping the entire shot to a small number of representative frames, called *key-frames* [14]. Indeed, an index may be constructed from key-frames, and retrieval may be directed at key-frames, which can subsequently be displayed for browsing purposes.

This paper focuses on the key-frame extraction techniques. There exist many different approaches to extract key-frames [14] [15] [1]. However, they can not effectively capture the major visual content, and/or are not friendly-user where a set of parameters must be adjusted by the user, and/or also are computationally expensive.

In this paper, a new strategy to extract the most characteristic key-frames is proposed. The main idea is to cluster similar or redundant views within the shot together. The clusters are approximated by a mixture of Gaussians using the standard Expectation-Maximization (EM) algorithm [4]. Here, the estimation is performed in the color histogram feature space. The Bayes Information Criterion [12] is used to chose the appropriate number of clusters (i.e. the number of key-frames) for each shot differently, depending on its complexity. From each obtained cluster, firstly the closest frame to the median of its frames is selected as a reference key-frame. Then depending on the variation in time and appearance of the cluster content against the reference key-frame multiple frames can be extracted to represent effectively the cluster. A temporal filter is applied on the set of all selected key-frames in order to eliminate the overlapping case between constructed clusters of frames. Using the proposed framework only sufficient separated frames in time and appearance are kept. The selection of key-frames is fully automatic, no parameters to be adjusted by the user.

The organization of this paper is as follows. Sections 2 and 3 review and evaluate respectively some relevant approaches to the present work. Section 4.1 details the clustering strategy and section 4.2 describes the algorithm to extract key-frames. In this work the key-frames are extracted for only browsing purposes since key-frames summarize the content of a shot [9]. In section 5 experimental results on different tracked objects in a real-world video sequence are presented. The video sequence has been already segmented into shots [3] and moving objects are localized and tracked in shots [5]. These experiments demonstrate the performance of the proposed technique. A short discussion and concluding remarks are given in sections 5.2 and 6 respectively.

2 Related work

Many research effort have been given in the area of key frame extraction [14] [15] [1] [13]. They could be regrouped in three following categories.

1. *Shot boundary based approach.* O'connor et al. use either of the first, the middle or the last frame of the shot as the shot's key frames [11].

2. *Motion analysis based approach.* Wolf proposes a motion based approach to key-frame extraction [13]. He first computes the optical flow for each frame and then computes a simple motion metric based on the optical flow. Finally he analyzes the metric as a function of time to select key-frames at the local minima of motion.

3. *Visual content based approach.*

   - Zhang et al. propose to use color and motion features independently to extract key frames [14]. The similarity between the current frame and the last key-frame is identified in each feature space by a thresholding technique.
   - Motivated by the same observation as Wolf's and Zhang Avrithis et al. combine the color and motion features in a fuzzy feature vector [1]. The trajectory of

feature vectors of all frames of a given shot is analyzed firstly. Then the key-frames are selected on the curve points: the local minima and maxima of the magnitude of the second derivative on the initial trajectory, in the discrete case.

- Zhuang et al. propose an adaptive key frame extraction using a linear clustering technique to regroup similar frames together [15]. The similarity between images of the same shot is computed in the 128-dimensional Hue-Saturation color histogram space. Based on a predefined threshold of similarity for each video sequence, the number of clusters is determined. After that, an arbitrary point of each cluster is selected as a key-frame. Only clusters of proportions greater than a predefined threshold are represented.

3 Evaluation of existing techniques

The approach of O'connor is the easy way to extract key frames. However, it does not capture the visual content of the video shot. The methods of Avrithis and Wolf give interesting results. However they are computationally expensive due to their analysis of motion, and their underlying assumption of local minima does not work very well in the case of constant variation of the feature vectors. The methods of Zhuang and Zhang are relatively fast. However, they are very sensible to the choice of the threshold of similarity. As a result, the number of selected key-frames is very variable. The adjustment of the threshold parameter represents a challenging problem for the user of these methods.

Next section details the theoretic part of the proposed framework to automatically select the effective key-frames for a given shot. Our approach uses an unsupervised clustering algorithm to group similar frames within a shot together. The Gaussian mixture density is used to model the temporal variation of color histograms in the RGB color space. In order to select automatically the number of appropriate components (clusters) the Bayes Information criterion is performed.

4 Probabilistic framework for shot abstraction

Assume that temporal video segmentation into shots was already performed. Then, each frame within a shot a is characterized by a vector of measurements called *feature*. Each feature is represented by a single *point* or *individual* in the d-dimensional feature space, where its coordinates are the values of the feature vector.

Now, for a given shot of n frames (or a tracked object of n occurrences), n points in the d-dimensional space describe the temporal variation (trajectory) of its contents. For example, figure 1 illustrates the temporal variation of the tracked "Ford Car" within a video shot of 66 frames. Some images of this shot are depicted in figure 3. Each point represents a RGB histogram computed on an occurrence of the tracked car in the shot. The 64-dimensional space of this data was already reduced performing the Principal Components Analysis (PCA). In the current framework the method of [5] was used to track non-rigid objects.

In the following both clustering strategy to classify similar frames together, and key-frames extraction algorithm to realize the abstraction level are described.

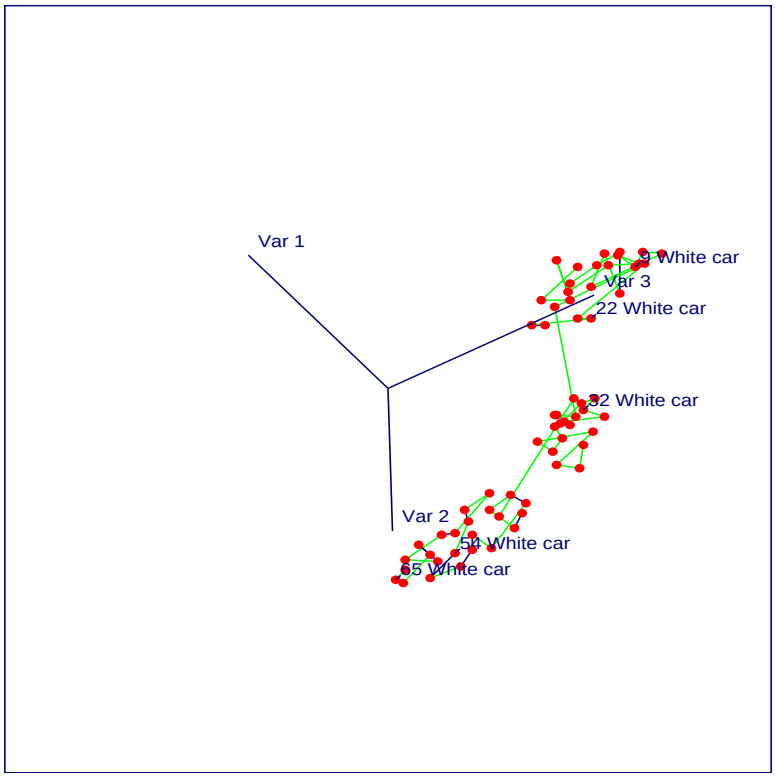


Figure 1: Illustration of the content-based variation of the tracked "Ford Car" within the shot of figure 3, in the 3-principal components of the RGB histogram space. Some labels of points are shown (e.g the corresponding number of frames).

4.1 Clustering by Gaussian mixture densities

Again, assume that a video shot consisting of n images has been selected. Let us denote by y_i the feature vector of dimension d that characterizes the ith frame, and by $Y = \{y_i;\ i = 1...n\}$ the set of feature vectors collected for all frames of the shot. The distribution of Y is modeled as a joint probability density function, $f(y \mid Y, \theta)$ where θ is the set of parameters for the model f. We assume that f can be approximated as a J-component mixture of Gaussians [10]:

$$f(y|\theta) = \sum_{j=1}^{J} p_j \varphi(y|\alpha) \tag{1}$$

where the p_j's are the mixing proportions and φ is a density function parameterized by the center and the covariance matrix, $\alpha = (\mu, \Sigma)$. In the following, we denote $\theta_j = (p_j, \mu_j, \Sigma_j)$, for $j = 1, \ldots, J$ the parameters to be estimated.

Each cluster approximated by a Gaussian component of the mixture groups a set of similar points (i.e. similar frames) in the feature space. Thus a transition from one Gaussian component to another indicates a significant temporal variation within the shot.

Parameters Estimation. Gaussian mixture density estimation is performed in a semi-parametric way so that the number of components scales with the complexity of the data and not with the size of the data set. The density estimation procedure is a missing data estimation problem to which the EM algorithm [4] can be applied. The type of Gaussian mixture model to be used (see next paragraph) has to be fixed and also the number of components in the mixture. If the number of components is one the estimation procedure is a standard computation (step M), otherwise the expectation (E) and maximization (M) steps are executed alternately until the log-likelihood of θ stabilizes or the maximum number of iterations is reached.

Let $\mathbf{y} = \left\{y_i;\ 1 \leq i \leq n \ and \ y_i \in I\!R^d\right\}$ be the observed sample from the mixture distribution $f(y|\theta)$. We assume that the component from which each y_i arises is unknown, so that the missing data are the labels c_i $(i = 1, \ldots, n)$. We have $c_i = j$ if and only if j is the mixture component from which y_i arises. Let $\mathbf{c} = (c_1, \ldots, c_n)$ denote the missing data, $\mathbf{c} \in B^n$, where $B = \{1, \ldots, J\}$. The complete sample is $\mathbf{x} = (x_1, \ldots, x_n)$ with $x_i = (y_i, c_i)$. The complete log-likelihood is

$$L(\theta, \mathbf{x}) = \sum_{i=1}^{n} \log \left\{ \sum_{j=1}^{J} p_j \varphi(x_i|\mu_j, \Sigma_j) \right\}. \tag{2}$$

The EM algorithm at iteration "m" is summarized as follow :

Step-E : For $i = 1, \ldots, n$ and $j = 1, \ldots, J$ compute the conditional probability, given $\mathbf{y}$, that y_i arises from the mixture component with density $\varphi(.|\mu_j^m, \Sigma_j^m)$ and mixing proportion p_j^m

$$t_{ij}(\theta^m) = \frac{p_j^m \varphi(x_i, \mu_j^m, \Sigma_j^m)}{\sum_{\ell=1}^{J} p_\ell^m \varphi(x_i|\mu_\ell^m, \Sigma_\ell^m)}. \tag{3}$$

Step-M : Maximize the log-likelihood conditionally on t_{ij}^m. Indeed, in the case of a general Gaussian model we get for θ^{m+1}

$$\begin{aligned} p_j^{m+1} &= \frac{1}{n} \sum_{i=1}^{n} t_{ij}(\theta^m) \ ; \ \mu_j^{m+1} = \frac{\sum_{i=1}^{n} t_{ij}(\theta^m)\ y_i}{\sum_{i=1}^{n} t_{ij}(\theta^m)} \\ \Sigma_j^{m+1} &= \frac{\sum_{i=1}^{n} t_{ij}(\theta^m)(y_i - \mu_j^{m+1})(y_i - \mu_j^{m+1})^\intercal}{\sum_{i=1}^{n} t_{ij}(\theta^m)}. \end{aligned} \tag{4}$$

At each iteration, the following properties hold: For $i = 1, \ldots, n$

$$\sum_{j=1}^{J} t_{ij}(\theta^m) = 1 \text{ and } \sum_{j=1}^{J} p_j^m = 1. \quad (5)$$

More details on the EM algorithm could be found in [4]. Initialization of the clusters is done randomly. In order to limit dependence on the initial position, the algorithm is run several times (10 times in our experiments) and the best solution is kept.

Gaussian models. Gaussian mixtures are sufficiently general to model arbitrarily complex, non-linear distribution accurately given enough data [4]. When the data is limited, i.e. the number of frames of a shot is small, the method should be constrained to provide better conditioning for the estimation. For these reasons and in order to make the method fast some constraints are added on the covariance parameter. In a previous work we have described these Gaussian models and their application [8]. These models are basically introduced in [4].

Choosing models and mixture components' number. To avoid a hand-picked number of Gaussians in the mixture, i.e. the number of clusters and then the number of key-frames to be selected, the Bayes Information Criterion (BIC) [12] is used to determine the best probability density representation (appropriate Gaussian model and number of components). It is an approach based on a measure that determines the best balance between the number of parameters used and the performance achieved in classification. It minimizes the following criterion:

$$BIC(M) = -2L_M + Q_M ln(n) \quad (6)$$

where L_M is the maximized log-likelihood of the Gaussian model M and Q_M is its number of free parameters.

4.2 Key-frames extraction

A few images called "key-frames" can summarize the visual content of a video shot. By definition, a key-frame is an existing frame within the shot which represents a set of redundant similar frames (or views of objects). In addition, two key frames should be visually different.

This section details the extraction of key-frames algorithm for a given shot. As a result of the first part of the approach, a set of clusters are identified. Each cluster is characterized by its center (mass of the distribution), its covariance matrix (dispersion around the center) and the number of individuals belong to it.

The algorithm to extract key-frames from a given shot is of two stages:

- Perform the following procedure on each cluster C^k with $k = 1..K$ and K denote the number of constructed clusters of frames.

  1- Compute the *median frame*, F_m^k, for the set of frames $\{F_i^k; i = 1..n_k\}$ belong to the cluster C^k, n_k represents the proportion of cluster C^k.

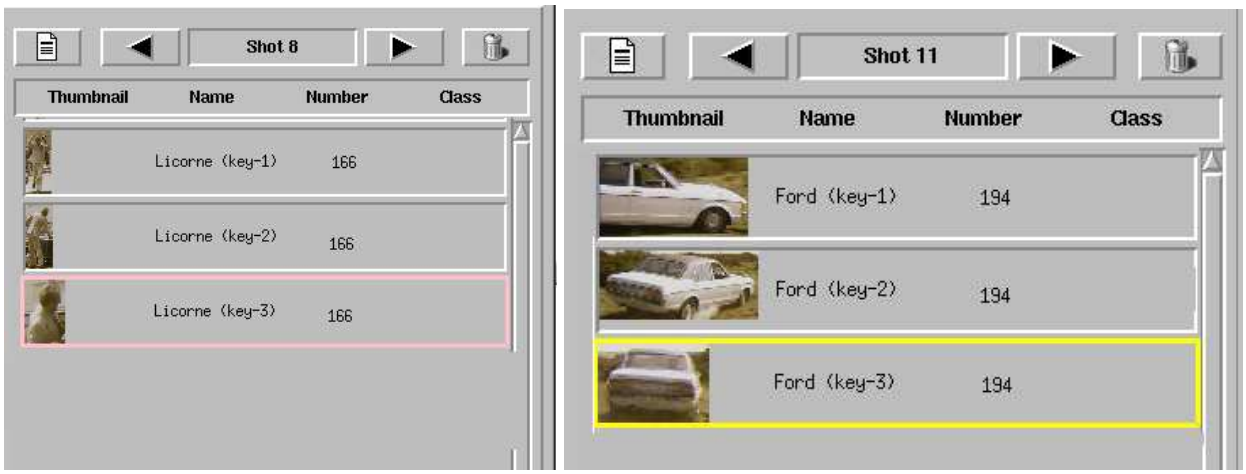


Figure 2: *Key frames browsing interface*

2- Select as a *reference key-frame*, F_r^k, the closest frame to F_m^k.

3- For each frame, F_i^k, belongs to C^k, check **(a)** if the temporal distance between it and the reference frames is greater than a predefined temporal threshold. If this condition is verified then check **(b)** if the similarity distance between it and the reference frames is greater than a predefined similarity threshold. If these two conditions are verified add this frame F_i^k to the set of *reference key-frames*. The Mahalanobis distance, $d_M(F_i^k, F_r^k) = (F_i^k - F_r^k)\Sigma_k^{-1}(F_i^k - F_r^k)^t$, is used here to compute the similarity between feature vectors of two frames.

- Merge the set of reference key-frames obtained for all clusters of a shot. From this set of frames keep only the key-frames which verify the two conditions (a) and (b) listed on the above procedure.

5 Implementations and experimental results

In our project for building and browsing interactive video [9] [7], a video sequence is segmented into shots first, using the method of [3], and then moving objects are localized and tracked in each shot separately. The method of [5] was used to track objects. As mentioned previously we extract the key-frames here for a browsing purposes. The browsing of key-frames allows a fast visualization of the content of the shot (see figure 2 for example).

In the current experiments each occurrence of a tracked object is characterized by a histogram computed in the RGB color space. The histogram approach is well known as an attractive method for image retrieval because of its simplicity, speed and robustness. The RGB space is quantized into 64 colors. Then, the Principal Component Analysis was applied on the entire set of vector features in order to reduce their dimensionality. Only 10

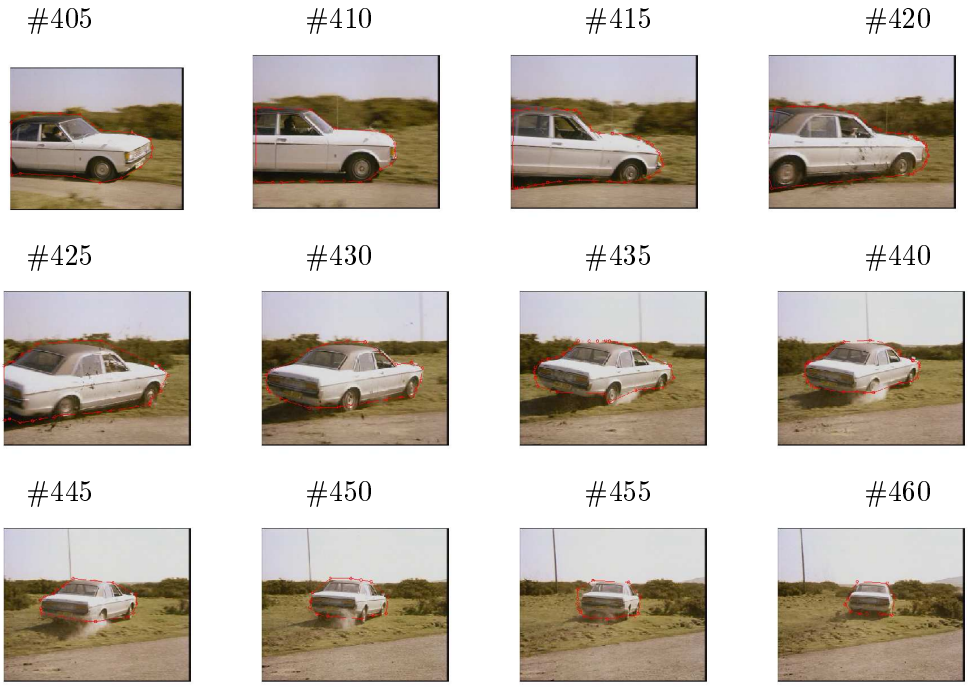


Figure 3: *Subset of views of the "Ford Car" sequence of* 66 *frames.*

eigenvectors are kept corresponding to the 10 largest eigenvalues. Thus, in this new space the clustering strategy was applied. This makes the method more accurate and speed.

5.1 Results

Experiments are conducted on the MPEG "Avengers" TV movie of "Institut National de l'Audiovisuel en France" (INA). The extraction of key-frames is performed separately on each tracked object within a shot. During the estimation process, the maximum number of permitted Gaussian components, K, depends on the number of frames in the shot. Using the BIC criterion, the appropriate number of cluster ($\in [1..K]$) is chosen automatically i.e the the number of selected key-frames. The size of the temporal window was fixed to 20 frames which is reasonable to separate two key-frames in time.

Figure 4: Key-frame results for the "Ford Car" sequence

To evaluate the effectiveness and accuracy of the proposed key-frame extraction technique, we illustrate in this paper the result on two different tracked objects of the database. The "Ford Car" and "la Licorne" sequences, consisting of 66 and 100 frames are illustrated in figures 3 and 5 respectively. One every 5 frames is depicted. The results of the proposed approach are presented in figures 4 and 6.

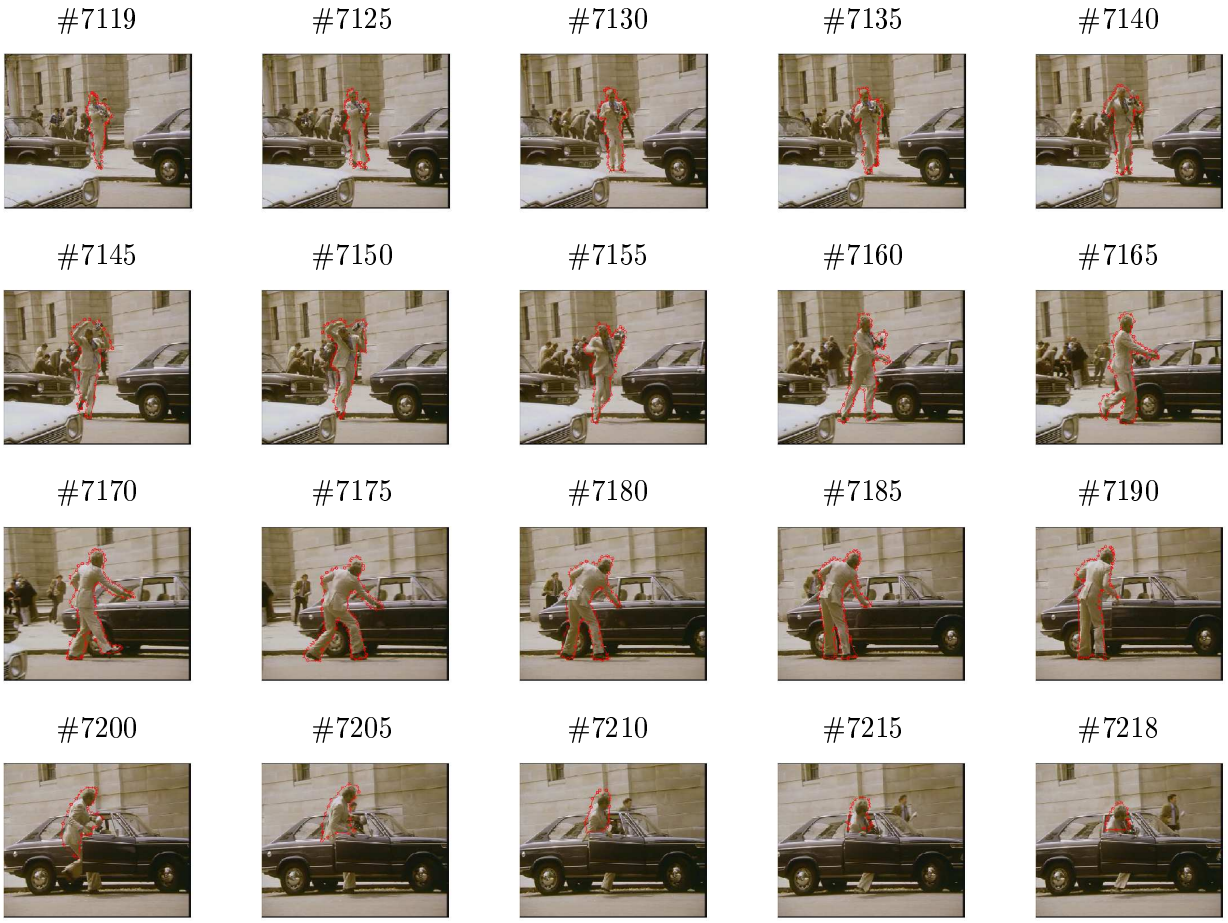


Figure 5: Subset of views of the "la Licorne" sequence of 100 frames.

5.2 Discussion

For each experimented shot, it can be seen that the selected key-frames provide sufficient visualization of the total frames of the shot. They are clearly representative of the different views of tracked objects which are continuously changing with time.

The closest work to our approach is the work of [15]. Zhuang et al. use a linear clustering algorithm with a predefined threshold to determine the number of clusters. Then, they represent each formed cluster of frames by an arbitrary one. The technique presented here uses the EM algorithm which is more adequate to find the partition of

Figure 6: Key-frame results for the "la Licorne" sequence

a complex distribution where the number of clusters (complexity of the distribution) is determined automatically using the BIC criterion. Also, the key-frames are extracted here for tracked objects.

It is obviously that the method of Zhuang et al. is more speed because the employed clustering strategy is linear and non-parameterized. The proposed approach here is adopted by our project [9] since the estimated Gaussian components are used before the selection of key-frames, by the recognition process of similar tracked objects in the whole video sequence [7].

6 Conclusion

In this paper, an efficient video content representation has been presented for realizing an abstraction level beyond the shot one: the key-frame level. The presented framework represents a full automatic method to extract key-frames where no parameters are needed to be adjusted by the user. The use of the PCA and the addition of constraints on the covariance matrix make the method relatively fast.

The experiments on different real-world tracked objects shown the performance of the proposed technique. Such a technique can be performed on non-segmented frames of shots. It is able to capture the salient visual content of the key clusters and thus that of the underlying shot. The color histogram was computed as a feature where another features could be tested. However, accuracy of the estimation of the Gaussian mixture densities is related to the dimension of the feature space which must be chosen carefully in respect to the size of a shot.

The key-frames are extracted here for a browsing purposes only, but an index may be constructed from key-frames, and retrieval may be directed at key-frames. Finally, the estimated Gaussian models of tracked objects can be used to recognize similar objects in the whole video. A work on this research point is in progress.

Acknowledgments

We would like to acknowledge Alcaltel CRC for its support of this work, and the "Institut National de l'Audiovisuel en France", dept of Innovation, for providing the video used in this paper.

References

[1] Y. S. Avrithis, A. D. Doulamis, N. D. Doulamis, and S. D. Kollias. A stochastic framework for optimal key frame extraction from mpeg video databases. *Computer Vision and Image Understanding*, 75(1/2):3–24, July-August 1999.

[2] J. S. Boreczky and L. A. Rowe. Comparison of video shot boundary detection techniques. In *Proc. SPIE Conf. on Vis. Commun and Image Proc.*, 1996.

[3] P. Bouthemy and F. Ganansia. Video partitionning and camera motion characterisation for content-based video indexing. *Proc. 3rd IEEE Int. Conf. Image Processing.*, september 1996.

[4] G. Celeux and G. Govaert. Gaussian Parsimonious Models. *Pattern Recognition*, 28(5):781–783, 1995.

[5] M. Gelgon and P. Bouthemy. A region-level graph labeling approch to motion-based segmentation. In *CVPR*, pages 514–519, Puerto Rico, June 17-19 1997.

[6] N. Guimaraes, N. Correia, I. Oliveira, and J. Martins. Designing computer for content analysis: A situated use of video parsing and analysis techniques. *Multimedia Tools and Applications*, 7:159–180, 1998.

[7] R. Hammoud and R. Mohr. Building and browsing hyper-videos: a content variation modeling solution. *Pattern Analysis and Applications*, 2000. Special Issue on Image Indexation, Submitted.

[8] R. Hammoud and R. Mohr. Gaussian mixture densities for indexing of localized objects in a video sequence. Technical report, INRIA, March 2000. http://www.inria.fr/RRRT/RR-3905.html.

[9] R. Hammoud and R. Mohr. Interactive tools for constructing and browsing structures for movie films. In *ACM Multimedia*, pages 497–498, Los Angeles, California, USA, October 30 - November 3 2000. (demo session).

[10] R. Hammoud and R. Mohr. Mixture densities for video objects recognition. In *International Conference on Pattern Recognition*, volume 2, pages 71–75, Barcelona, Spain, 3-8 September 2000.

[11] B.C. O'Connor. Selecting key frames of moving image documents : A digital environment for analysis and navigation. *Microcomputers for Information Management*, 8(2):119–133, 1991.

[12] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, March 1978.

[13] W. Wolf. Hey frame selection by motion analysis. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, 1996.

[14] H. J. Zhang, C. Y. Low, S.W. Smoliar, and J.H. Wu. Video parsing, retrieval and browsing: An integrated and content-based solution. *ACM Multimedia*, pages 15–24, 1995.

[15] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *Proc. of IEEE conf. on Image Processing*, pages 866–870, Chicago, IL, October 1998.

Chapitre 7

Construction des scènes pour un film vidéo

Je vais me donner de la marge
côté délais.

Les images successives d'un film prises par la même caméra et sans interruption temporelle sont regroupées dans des *plans vidéo*. A un niveau plus sémantique dans la structure hiérarchique des films (figure 2.1) les plans successifs décrivant une continuité d'action et de lieu peuvent être regroupés dans des *scènes*. Comme nous l'avons déjà mentionné dans l'introduction de ce mémoire, ce niveau permet à l'utilisateur final de la vidéo hyperliée d'explorer un film comme un livre avec une table de matières.

Au contraire de l'identification automatique des ruptures des plans basée sur une simple comparaison des couples des images successives, l'identification des scènes est une tâche très délicate. Ceci est dû d'une part à l'inexistence d'une formulation précise de la rupture entre deux scènes (surtout lorsque l'image est analysée seulement) et d'autre part à la difficulté d'une caractérisation fiable du contenu des plans et de la mise en oeuvre d'une stratégie de classification automatique des plans.

Ce chapitre se focalise sur la construction de scènes pour un film vidéo. D'abord on rappelle les étapes fondamentales de l'approche de macro-segmentation que nous avons proposée lors d'un travail de stage de DEA [51] [53] [52] [25]. Ensuite nous résumons l'extension de cette approche élaborée durant cette thèse [54]. Cette extension porte essentiellement sur la manière de comparer deux plans vidéo sur la base de plusieurs descripteurs.

7.1 Deux étapes pour la macro-segmentation en scènes

La motivation principale de la macro-segmentation est dûe au fait qu'un film vidéo d'une heure par exemple peut contenir plus que 1000 plans, ce qui est difficile à représenter

graphiquement et à explorer linéairement par l'utilisateur. Le découpage d'un film vidéo en des unités plus macroscopiques que les plans, mais aussi sémantiques, sera appelé découpage en *scènes*, avec la définition imprécise suivante: une scène est un ensemble de plans adjacents qui comporte une certaine unité de lieu d'acteur, d'action et/ou du son.

La macro-segmentation peut-être vue comme un processus qui consiste à créer une partition de l'ensemble de tous les plans d'un film vidéo, à condition que chaque partie formée soit continue dans le temps – une scène –.

De là, une première étape de classification des plans similaires dans des blocs et une seconde étape de fusionnement des blocs qui se chevauchent dans le temps jusqu'à la vérification de la continuité temporelle, permettent de générer des scènes. Ces deux étapes ont été adoptées dans l'approche que nous avons proposée dans un travail antérieur sur la macro-segmentation. Nous décrivons brièvement ces deux étapes ci-dessous et pour plus de détail le lecteur pourra se reporter aux références [53] [25].

Procédure Macro-segmentation, deux étapes :

1) *Fabrication des blocs* ou de classes de plans. Les plans sont représentés par les images médianes, les histogrammes de couleurs sont extraits de ces images et un algorithme adaptatif de classification est employé pour les regrouper en blocs. A chaque itération de cet algorithme un plan non étiqueté constitue un nouveau bloc ou bien il est classé dans un bloc déjà existant selon une distance de similarité et un seuil fixé a priori. Un algorithme similaire a été employé plus tard par Zhuang [134] pour la sélection des images clés.

2) *Construction de scènes* par fusion de blocs en relation. A partir de la répartition des plans des différents blocs sur l'axe de temps, quatre relations temporelles sont générées entres ces blocs: *Before*, *Meets*, *Overlaps* et *During*. Une relation *Before* décrit une transition entre deux séquences narratives et elle est générée entre deux blocs dont le dernier plan du premier bloc (ordre chronologique) est lié par un effet de transition progressive avec le premier plan du deuxième bloc. Par contre si le raccord entre ces deux plans est de type "cut" alors une relation de type *Meets* est générée. Deux scènes d'une même séquence narrative sont reliées par une relation de type *Meets*. Lorsque deux blocs se chevauchent dans le temps une relation temporelle de type *Overlaps* ou *During* est générée. Deux blocs d'une même scène sont reliées par une relation de type *Overlaps* ou *During*. La figure 7.1 illustre un graphe temporel de blocs pour les 40 premiers plans de la séquence "Dances with Wolves". La construction des scènes se fait donc par fusions successives des blocs reliés par des relations de type *Overlaps* et *During* jusqu'à la vérification d'un critère de continuité temporelle et la détection des relations *Meets* qui déconnecte le graphe temporel de blocs ainsi construit.

Cette approche a donné des résultats satisfaisants sur la séquence "Dances with wolves" comportant environ 10000 images et segmentée en 70 plans par la méthode de [4] décrite dans la section 2.3.2. Les figures B.1 et B.2 (annexe) illustrent les images-médianes de ces plans. Cinq scènes sont identifiées correctement : première scène du plan 1 à plan 27,

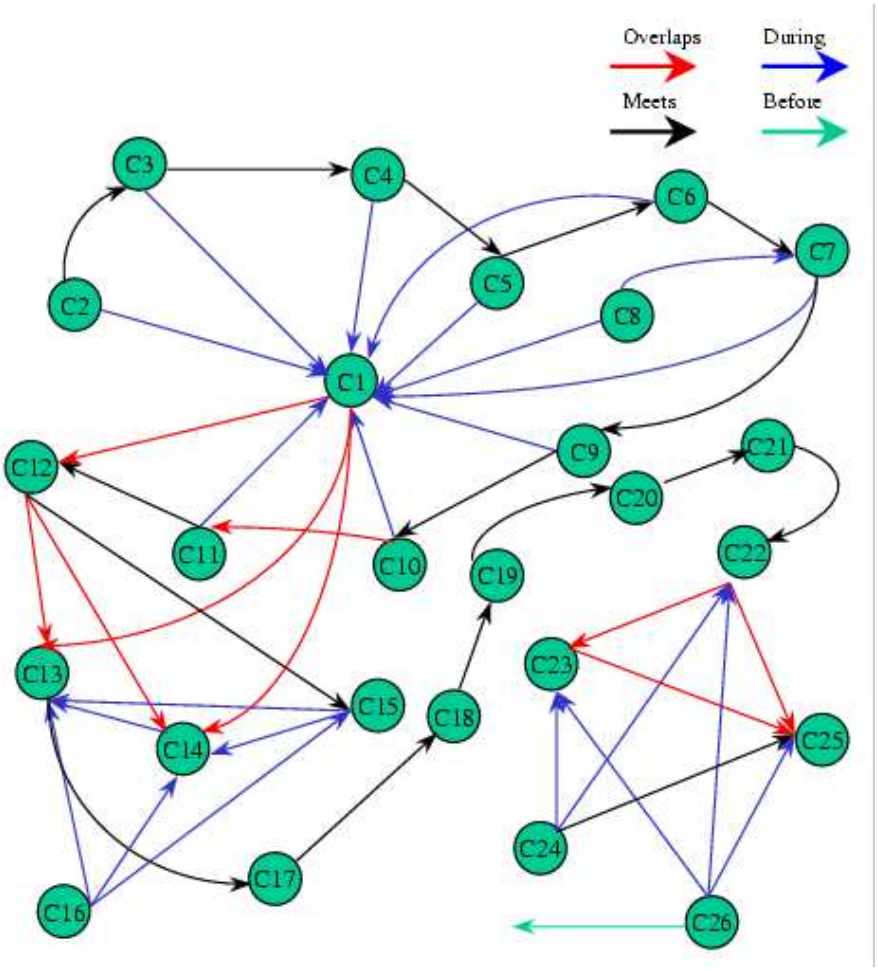


FIG. 7.1: *Exemple d'un graphe temporel de blocs généré sur les 40 premiers plans du film "Dances with wolves"; chaque noeud représente un bloc de plans similaires; la similarité est calculée sur la base des histogrammes des images médianes des plans.*

deuxième scène comporte le plan 30 seulement, troisième scène du plan 31 à plan 47, quatrième scène comporte le plan 52 seulement et cinquième scène est composée du plan 53 à plan 67. Les plans restants 28, 29, 31, 32, 48, 50, 51, 68, 69 et 70 sont considérés comme des scènes indépendantes de taille 1. Celles-ci représentent les fausses détections des scènes produites par l'approche.

Dans cette approche seule l'unité de lieu (histogramme global de couleurs) a été considérée dans la fabrication de blocs de plans. Cette restriction peut expliquer le fait d'obtenir des scènes de taille 1. Autrement dit, l'unité de lieu n'est pas toujours capable d'identifier deux plans de la même scène. D'autres indices de comparaisons des plans doivent être pris en considération. Dans ce sens nous discutons l'extension de cette approche dans la section suivante.

Dans le même contexte de travail, Yeung et al. [130] ont adoptés une technique similaire à celle décrite ci-dessus. En se basant sur nos travaux, Mahdi, Chen et Ardebilian [81] [80] ont proposés plusieurs extensions en introduisant la notion du *rythme* des plans successifs et en rajoutant une phase d'identification des plans d'intérieurs et d'extérieurs. Une autre approche de macro-segmentation basée sur des règles déduites des effets de montage a été proposée par Aigrain et Joly [1]. Ils utilisent 10 règles d'identification des ruptures de séquences de scènes dans un film. Mais cette approche paraît très restrictive compte tenu du foisonnement des indices de ruptures de scène qui se trouvent dans différents films. Par exemple cette méthode segmenterait le film "Titanic" (2h50) en trois scènes dont les limites correspondent aux effets de montage réalisés lors des transitions où Rose (Kate Winslett) commence une phase de la narration de son histoire. Enfin, une approche très récente proposée par Sundaram et al. [118] consiste à coupler la bande image et la bande sonore semble être bien adaptée à ce problème d'identification de scènes.

7.2 Résumé de "A mixed classification approach of shots for constructing scene structure for movie films" – IMVIP'2000

Cet article, publié à IMVIP'2000, présente une approche mixte de macro-segmentation de la vidéo en scènes. L'idée ici est d'étudier la similarité entre deux plans vidéo sur la base de plusieurs unités ou descripteurs de bas niveaux. Le problème issu de cette extension de l'approche de base que nous avons présentée dans la section précédente est la fusion de ces descripteurs. Cette phase de fusion des descripteurs devra produire une partition unique et fiable de l'ensemble de tous les plans d'un film vidéo et permet ensuite d'appliquer la deuxième étape de "construction de scènes".

Le contenu visuel d'un plan revêt une masse importante d'informations qui permet normalement de le distinguer ou de l'associer à d'autres plans. Plusieurs autres descripteurs que l'histogramme de couleurs peuvent être utilisés pour caractériser un plan. C'est le cas par exemple des objets identifiables à l'intérieur de chaque plan, de la position relative de ces objets dans chaque plan, des indices indiquant le décor où les images ont été tournées, etc.

Dans notre implémentation, seules l'unité de lieu représentée par l'histogramme et

l'autocorrélogramme de couleurs et l'unité de contenu du plan représentée par les objets localisés dans les plans ont été utilisées. L'approche est générale et d'autres descripteurs peuvent être testés. La similarité entre deux plans vidéo sera donc calculée ici sur la base des histogrammes de couleurs, des autocorrélogrammes (voir section 3.2.2.1) et du nombre d'objets similaires contenus dans les plans. Dans ces expérimentations, les objets sont caractérisés par des autocorrélogrammes. Un seuil de similarité a été fixé a priori pour chaque type de distances entres descripteurs. Pour montrer l'importance de cette caractérisation mixte du plan vidéo, considérons l'exemple des plans de la figure 7.2. Chaque plan est représenté par son image médiane. Certaines images contenants d'objets d'intérêts ont été segmentées manuellement. Les plans des couples (7, 9), (33, 50) et (69, 70) appartiennent à une même scène du film "Dances with Wolves" (voir figure B.1). Les deux plans 7 et 9 sont détectés similaires sur la base des histogrammes et des autocorrélogrammes et différents avec le descripteurs de contenu. Le plan 9 ne contient pas des objets d'intérêts. Par contre, les deux plans 33 et 50 sont reliés sur la base de leurs contenus et non pas par les descripteurs globaux. Enfin, les deux derniers plans de la figure 7.2 ont été appariés correctement par les histogrammes de couleurs, mais échec avec les descripteurs de contenu et les autocorrélogrammes.

L'exemple cité plus haut montre que chaque descripteur permet de comparer les plans suivant une caractéristique particulière. Les histogrammes de couleurs tiennent compte de la distribution globales de couleurs, les corrélogrammes tiennent compte des informations locales dans l'image et des environnements de tournage, le contenu des plans permettent de comparer les plans suivant le contenu.

En traitement d'images la comparaison de deux images sur plusieurs points de vue est un problème difficile. Une solution souvent adopté pour ce problème consiste à déduire une distance unique par combinaison des distances des différents attributs. Bisson [11] a proposé une mesure de similarité entre objets dans des modèles à objet qui tiennent compte aussi bien des attributs de chaque objet que des relations qui lient les objets à comparer. Valtchev [123] utilise une distance dérivée de la distance de Mahalanobis pour la construction automatique des taxonomies pour les langages à objets. La distance entre deux images est considérée comme la somme des écarts obtenus sur les attributs et les relations. Jain et al. proposent d'utiliser le barycentre des distances pour la combinaison de la couleur et de la forme lors de la comparaison des images [67].

La méthode de Jain et al. est très simple à mettre en oeuvre mais est-il possible de trouver une pondération entre des entités qui n'ont rien en commun ? Chaque mesure de similarité illustre un aspect particulier de la vidéo, des aspects indépendants les uns des autres. Une esquisse de cette approche a été utilisée dans un premier temps pour la classification des objets dans la vidéo, avec résultats nullement satisfaisants et très sensibles à la variation des coefficients de pondération des distances [35].

Une combinaison linéaire simple des distances entre les plans ne peut donc pas être utilisée pour effectuer une classification automatique des plans. C'est dans cet optique qu'une méthode de construction des blocs par fusion, non pas des distances suivant chaque descripteur, mais des blocs de plans obtenus à partir de chaque descripteur a été proposée. Dans un premier temps nous avons défini une nouvelle distance entre les blocs basée sur l'intersection des ensembles de blocs de plans fabriqués pour chaque descripteur de la vidéo.

FIG. 7.2: *Similarité entre deux plans caractérisés par trois descripteurs : histogramme de couleurs, autocorrélogramme, et le contenu. Une distance pour chaque type de descripteur est calculée entre les plans (7, 9), (33, 50) et (69, 70). Les plans de chacun de ces couples appartiennent à une même scène du film "Dances with Wolves" (voir figure B.1). Les résultats d'appariements par ces trois descripteurs sont les suivants : les plans* 7 *et* 9 *sont identifiés sur la base des histogrammes et des autocorrélogrammes; les plans* 33 *et* 50 *sont correctement appariés sur la base de descripteurs de contenu seulement; et la similarité des plans* 69 *et* 70 *est détectée seulement sur la base des histogrammes de couleurs.*

Dans un second temps, nous avons proposé un algorithme de fusion de deux étapes: une étape de fusion verticale qui permet la comparaison et la fusion des ensembles construits à partir des descripteurs différents, et une étape de fusion horizontale qui permet un affinage des résultats obtenus à partir de la première étape. En effet, les ensembles de blocs obtenus après la première étape ne sont pas disjoints et l'étape d'affinage les corrige.

A la première étape, le seuil de similarité entre deux plans est choisi empiriquement mais il est strict. Ce seuil strict implique une forte similarité entre les plans regroupés suivant un descripteur donnée. A titre d'exemple, ce seuil est fixé à 0.04 pour les histogrammes de couleurs alors que dans [53] on utilise un seuil de 0.09 sur la séquence "Dances with Wolves". Ce seuil strict donne l'assurance de rejeter les mauvais appariements suivant un descripteur donnée. Les rejets occasionnés par un descripteur donné pourront être corrigés par les autres descripteurs lors de l'opération de fusion.

La construction des blocs de plans à l'aide de cet algorithme de fusion présente l'avantage de tirer le maximum d'informations de chaque descripteur lors de la comparaison de plans. Cette méthode nous a permis de générer des blocs sur des séquences vidéo malgré une forte variabilité dans le type de leur contenus.

L'application de cette approche sur la séquence "Dances with Wolves" a donné des résultats parfaits: les sept scènes de la séquence sont correctement identifiées (voir annexe B). Sur d'autres séquences du film "Avengers", dont leurs contenus sont très complexes, les résultats obtenus sont satisfaisants et nettement meilleurs que lorsqu'un seul descripteurs est utilisé.

La figure 7.3 illustre les cinq scènes de la séquence Avengers obtenues par l'approche mixte. L'évaluation précise de la qualité de cette segmentation est une tâche délicate car la séparation entre les scènes de la séquence traitée n'est pas bien définie. Une segmentation manuelle de cette séquence pourrait envisager de mettre le dernier plan de la première scène avec la deuxième ou bien de combiner la deuxième scène avec la troisième.

scène composée des plans [1..8] : “l’espion tire sur le prêtre”.

scène composée des plans [9..12] : “l’espion fuit et monte dans sa voiture”.

scène composée des plans [13..15] : “poursuite de deux voitures”.

scène composée des plans [16..23] : “discussion entre Steed, Prêtre et Purdey”.

scène composée de plan 24 : “décollage de l’avion”.

FIG. 7.3: *Les résultats de macrosegmentation de Avengers par l’approche mixte.*

A mixed classification approach of shots for constructing scene structure for movie films

R. Hammoud and D. G. Kouam

Irish **M**achine **V**ision and **I**mage **P**rocessing Conference, pages 223-230
September, 2000

Abstract: In order to facilitate the user's access to a movie film the scene structure should be constructed. A general framework for constructing scenes consists in clustering the shots into groups and then merging overlapped groups to extract scenes. The clustering of shots represents the most challenging task and should be done correctly. This paper presents a mixed approach to cluster shots using both features computed on key-frames and the corresponding localized moving objects. Two shots are matched on the basis of color-metric histograms, color-metric autocorrelograms and the number of similar objects localized into them. Using the hierarchical classification technique, a partition of shots is identified for each feature separately. From all these partitions a unified partition is deduced based on a proposed distance between their components (clusters). The components of the resulted partition are then linked together using the temporal relations of Allen in order to construct a temporal graph of clusters from which the scenes will be extracted. The experimental results of the proposed approach on three real-world video sequences demonstrate its performance. A comparative study between the original approach which uses only one descriptor and the extended one proposed here is analyzed and reported.

1 Introduction

In the context where the digital video data are available on the web in great quantity, the ability to understand and structure them becomes of greater importance, for the purpose of facilitating user's access to the video content. The automatic video structuring represents the fundamental task for many recently emerging multimedia applications: video browsing and navigation , content-based indexing and video summarization and trailers [17][7].

Commonly, two basis tasks for identifying the low-level structure of a video document are performed first: *shot boundary detection* and *key-frames extraction*. A *shot* is defined as an unbroken sequence of frames recorded from a single camera, which forms the building block of a video. Beyond the shot level an abstraction level can be constructed by mapping the entire shot to a small number of representative frames, called *key-frames* [17].

The drawbacks of low-level structures, shots and key-frames, are that they (1) contain too many entries to be efficiently presented to the user - for example there are 3225 shots in *October* of S.M. Eisenstein [12] to be presented to the user; and (2) do not capture the underlying semantic structure of the video based on which the user might wish to browse/retrieve [14] [16] [4].

In recent years the research in this area focuses on the construction of high level structures (*groups* and *scenes*) for movie films [14] [8]. The clustering of shots (or segmented objects in shots) creates links in the video ([6], [8]). In this case the access to the video is non-linear where the user can navigate in the content of a constructed group of non continuous shots (objects) in time. However, the *groups* level does not reflect the semantics of

the video. A *video scene* is defined as a collection of semantically related and temporally adjacent shots, depicting and conveying a high-level concept or story. The people watch the video by its semantic scenes not the physical shots or key-frames. The rest of this paper focuses on the construction of scenes for movie films.

Many approaches have been proposed in the literature for constructing scene structure [16] [15][4]. They basically perform two fundamental steps: "clustering of shots" into groups and "merging of overlapped groups" into scenes. The color histogram is commonly extracted from the representative key-frames of a shot. Based on this descriptor the visual similarity between shots is detected.

The clustering of shots represents the most challenge task of this work. It should be done correctly since the scene level is constructed on the top of the cluster one. It is obvious that two similar shots of the same scene do not have necessary the same global color distributions (histograms). Two different shots with the same global color distribution will be matched similar. On the other hand, the color histogram is very sensible to partial changes in the decor of a scene. In contrast, the content of shots, like localized objects, may be considered as important clues for detecting robustly similar shots.

Based on this short discussion, the similarity between shots is based upon many features. In this paper, a mixed approach to cluster shots using both features computed on key-frames and the corresponding localized moving objects. Two shots are matched on the basis of color-metric histograms, color-metric auto-correlograms and the number of similar objects localized into them. Using a hierarchical classification technique, a partition of shots is identified for each feature separately. From all these partitions a unified partition is deduced based on a proposed distance between their components (clusters). Following [4], the components of the resulted partition are then linked together using the temporal relations of Allen [1] in order to construct a temporal graph of clusters from which the scenes will be extracted.

The organization of this paper is as follows. The next section describes the recent works in this area. Section 3 details the proposed approach. Experimental results are given in section 4. A comparative study between the original approach which uses only one descriptor and the extended one proposed here is analyzed and reported in section 5. These experiments on three real-world video sequences demonstrate the performance of the proposed approach. Concluding remarks are given in section 6.

2 Related work

To construct the scene level beyond the shot one, Yeung et al. [16] identify first groups of shots using the hierarchical clustering algorithm, and then extract scenes form the *Scene Transition Graph* by merging overlapped clusters. Hammoud et al. [4] use an adaptive clustering algorithm to group shots. Then, they use the temporal relations of Allen (meets, before, overlaps and during) to link the formed clusters together. The linked clusters by "overlaps" and "during" relations are merged together. The "meets" relation between two non-merged clusters defines the boundaries of a scene. The "before" relation is used to describe the boundaries of a "narrative sequence". A "narrative sequence" is a set of

scenes of the film and it is detected when a gradual transition is identified. Both Yeung and Hammoud use the global color histogram to represent the visual content of a video shot (abstracted by a key-frame). Two shots are considered similar when the visual similarity between them is less than a predefined threshold and if they belong to the same predefined "temporal window" [16] or to the same "narrative sequence" [4].

Recently, Rui et al. [15] proposed an intelligent unsupervised classification technique to identify clusters of shots. At the shot level, they characterize the temporal information by extracting cumulated difference of histograms over all frames of the shot (called "shot activity"). The spatial information is summarized by the color histogram computed on the first and the last frames of shots. The distance similarities between shots, based on these descriptors, are normalized firstly, and then linearly combined using automatically determined thresholds. Our approach is close to this work by the use of multiple descriptors to characterize the video shot. But, the distance similarities based on these descriptors are used independently by the clustering process of shots. Then a unified partition of shots is deduced from all obtained partitions using a distance between their components (see the next section).

3 A mixed approach for constructing scene structure

The construction of the scene level requires a set of tasks to be done robustly: *basic segmentation of the video*, *characterizing of shots*, *clustering of shots* and *extraction of scenes*. In the following a description of these tasks is given.

3.1 Basic segmentation

To analyze a movie film a temporal segmentation into shots is done firstly. In this work we use the method of [2]; It relies on a robust, multi-resolution and incremental estimation of a 2D affine motion model between successive frames, accounting for the global dominant image motion. This method is also used to detect gradual transitions between shots (dissolves, black transitions, ...). These gradual transitions define the "narrative sequence" layer in the video.

Within a shot, the entities like moving objects can be detected and used later in the identification process of similar shots. Many research effort has been given in this area [10] [3]. The motion approach is widely used. In our approach, and following [3], the estimated motions during the *cut detection* process are used to localize and track mobile objects within shots.

3.2 Characterizing of shots

As mentioned in the introduction, a shot forms the building block of a video, and a cluster of similar shots forms the basic unit of a video scene. In this work, each shot is represented by only one key frame (the middle frame). However, more sophisticated approaches can be implemented [5], where multiple frames would be selected based on the density variation of the content of a shot.

The similarity between shots should not be necessary limited to the global color distribution. Two shots of the same scene may have very close contents like detected persons and very far global color distributions due to partial changes/occlusions in the decor of the scene. Figure 1 illustrates an example of two successive shots (represented by their middle images), of the same video scene, which have very close tracked objects (the yellow car) and totally different image backgrounds.

Figure 1: Two shots with similar tracked objects (yellow car) and different backgrounds

However, the global descriptors like color histograms and auto-correlograms [9] are useful for the recognition process when there is no detected objects in the framework of shots or when the appearances of the same object in different shots are very variable (partial occlusions, ...).

The color histogram captures only the color distribution in an image (or region) where the color auto-correlogram expresses how the spatial correlation of color changes with distance. Thus, the auto-correlogram is one kind of spatial extension of the histogram [9]. According to the type of information dominating in each shot, each descriptor allows a better comparison of a particular aspect of the shots, and consequently to lead to a comparison of the shots according to this aspect. In the presented framework, each video shot is characterized by its content (detected objects) and the color histogram and the color auto-correlogram both computed on the whole key frame.

3.3 Matching of shots

Two shots are matched on the basis of the global color histograms and color auto-correlograms via the L_1 distance measure. This distance is commonly used when comparing two feature vectors, because it is simple and robust.

$$L_1 = \sum_{i\in[1,n],k\in[d]} \frac{\mid h^{(k)}_{c_i 1} - h^{(k)}_{c_i 2} \mid}{1 + h^{(k)}_{c_i 1} + h^{(k)}_{c_i 2}} \tag{1}$$

where n is the total number of colors of an histogram and $h^{(k)}_{c_i j}$, $i = 1..n$, $j = 1, 2$, $k = 0$ or $k \in [1..d]$, represents the frequency of pairs of pixels of color c_i at a distance k in the jth image. For color histograms k is equal to zero.

In our experiments, the number of colors, n, was fixed to 64 and the spatial distance, d, used in the computation of auto-correlograms was fixed to 5 (this value of d is recommended by the author of this descriptor [9]).

Also, two shots are matched on the basis of the number of similar objects localized into them. The similarity between objects is determined using the above distance measured between auto-correlograms where a predefined threshold is determined in an interactive way.

Let S_i and S_j be the two shots of η_i and η_j objects detected into them respectively. Let η_{ij} the number of similar objects between S_i and S_j. The proposed metric distance to measure the number of similar objects between S_i and S_j is as follows:

$$D(S_i, S_j) = 1 - \frac{\eta_{ij}}{Max(\eta_i, \eta_j)} \tag{2}$$

For example, if S_i and S_j have the same number of objects and if these objects are matched similar, $D(S_i, S_j)$ will be close to zero, and so these two shots are considered similar according to their content.

3.4 Clustering of shots

By definition, a scene expresses an action of a short time duration and it belongs to a "narrative sequence" (see section 2). Based on this, a clustering strategy should take into account this temporal dimension and the boundaries of narrative sequences. Such considerations will avoid to classify two similar shots together if they are far in time [16] or/and if they do not belong to the same narrative sequence [4].

As described in the previous section, the matching of shots is done using three similarity distance measures. Our approach to cluster shots integrates these three distance measures and the above temporal criteria as follows.

Firstly, the complete-link hierarchical classification algorithm is performed on each proximity matrix of a similarity distance [11]. The number of clusters is determined using a predefined threshold of similarity; when the minimal distance between groups of shots is greater than the predefined threshold, the clustering process is stopped. Notice that, the threshold of similarity should be very close to zero in order to avoid a miss-classification of shots. This leads to a set of clusters with few number of elements. During the clustering process, the temporal distance between two grouped shots is checked as also if they belong to the same "narrative sequence". The "narrative sequences" are already identified during the temporal partitioning of the video into shots (see section 3.1) and the temporal threshold is determined in an interactive way.

Secondly, a unified partition of shots is deduced from the three partitions formed as described above. The proposed algorithm here to construct the final partition of shots consists in merging clusters of different partitions which have a certain number of common shots. It is based on the fact that the similarity between two shots is not always a linear combination of different descriptors, but this similarity may be reached using only one descriptor. Before to describe the two steps of this algorithm, we propose the following distance, $d_\cap$, to measure the intersection between two clusters of shots, ω_1 and ω_2:

$$d_\cap(\omega_1, \omega_2) = 2 - card(\omega_1 \bigcap \omega_2)(\frac{1}{card(\omega_1)} + \frac{1}{card(\omega_2)}) \tag{3}$$

The two parts of the mixed classification approach of shots are the following:
Clumping classification:

0- Let $\Omega_i, i = 1, ..., n$ be the n partitions constructed for the n different descriptors with $n \geq 2$. Initialize i to 1.

1- For each pair of clusters (ω_k, ω_l) that $\omega_k \in \Omega_i$ and $\omega_l \in \Omega_{i+1}$, if $d_{\cap}(\omega_k, \omega_l) \leq$ *Threshold*, then merge ω_l into ω_k and remove ω_l from Ω_{i+1}.

2- $\Omega_i = \Omega_i \bigcup \Omega_{i+1}$, $i = i + 1$. If $i < n$, goto 1, else goto 3.

Horizontal merging:

3- For each pair of clusters (ψ_k, ψ_l) that ψ_k and $\psi_l \in \Psi$ (the new constructed set of clusters), if $d_{\cap}(\psi_k, \psi_l) \leq$ *Threshold*, then merge ψ_l into ψ_k and remove ψ_l.

The *clumping classification* procedure produces a unique set of clusters of shots Ψ. However, some of these clusters are not disjoint. The *Horizontal merging* process consists in merging overlapping clusters together. The result is a partition of distinct clusters of shots.

3.5 Extraction of scenes

Generally, a scene/action of a movie film is projected as a continuous flow of shots in time. An intuitive way to construct the scenes is to adopt a strategy of merging clusters of shots as in [4] and [16]. In this section, an overview of the method of [4] to extract the scenes is given. This method will be used in our approach.

The shots are grouped into homogeneous clusters with respect to their temporal locality. The components of a cluster are not always successive in time and they are interleaved by the components of another clusters. Figure 2 (left) illustrates a representation of clusters on the time axis. The components of each clusters are displayed in their ascending order of the time code (frame number). Two shots may be linked by a simple "cut" or a gradual transition (GT). A gradual transition between two shots defines the boundaries of a narrative sequence. Notice that the searching of scenes will be done in each narrative sequence separately. Based on this representation of clusters versus the time, the sequential and parallel relations of Allen, Meets, Before, Overlaps and During, are generated between clusters for the purpose to link them [4]. The Overlaps and During relations are generated between two intersected clusters in time while the Meets relation links two successive clusters. When two clusters are successive in time but they belong to different narrative sequences a Before relation is generated.

At this stage, a temporal graph describes the movie film where the nodes are the formed clusters of shots and the edges represent the temporal relations between them. Figure 2 (right) illustrates an example of a temporal graph of clusters. The extraction of scenes is performed by exploring this temporal graph. Each pair of nodes linked by Overlaps or During relations are merged. This produces a new temporal graph of clusters where the edges are only Meets or Before relations and the nodes represent continuous

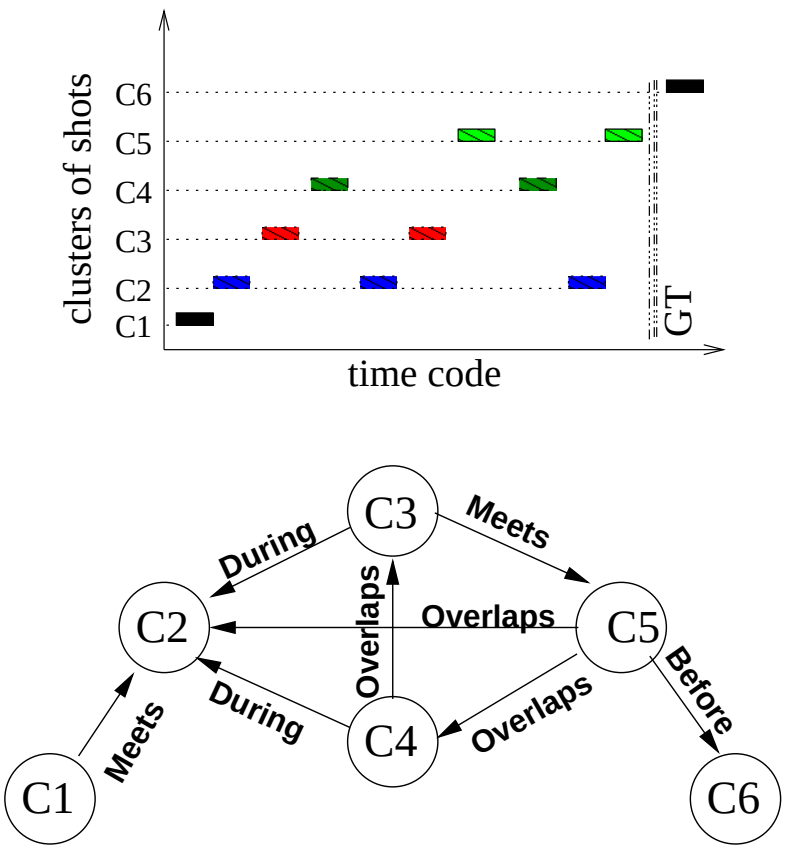


Figure 2: Representation of clusters of shots versus the time code (top) and the corresponding temporal graph of clusters (buttom).

blocks of shots in time. These blocks of shots define the scenes of a movie film. The Meets relations define transition between scenes of the same narrative sequence and the Before relations define transition between scenes of different narrative sequences.

4 Experimental results

The proposed approach has been experimented on three real-world video clips extracted from the *Avengers* and the *Dances with wolves* movie films, given by the "National Institute of Audiovisuel" in France (INA). The segmentation into shots and moving objects are done on the MPEG decompressed format of these videos. The experimental results are shown in table 1 where $\sharp Scenes$ denotes the number of scenes detected by the proposed approach. The number of frames, shots, segmented objects in key-frames, and the number of final clusters are denoted by $\sharp F_r$, $\sharp S_h$, $\sharp O_{bj}$ and $\sharp C_{lu}$ respectively. The evaluation of such a work is still a difficult task because there are no standard norms to define the boundaries of scenes. Following [15], the two measures of the effectiveness of the construction of scenes "false negatives" (false $\ominus$) and "false positives" (false $\oplus$) are also shown. The

"false negatives" indicates the number of scenes missed by the algorithm (for example, when two scenes are detected in a single one this measure is increment by one); and "false positives" indicates the number of scenes detected by the algorithm but which are not considered as scenes by human. These measures for scene's boundaries are obtained from subjective tests. Multiple human subjects are invited to watch the video clips and then asked to give their own structures. The structure that most people agreed with is used as the ground truth of the experiments.

Video name	♯					♯false	
	F_r	S_h	O_{bj}	C_{lu}	S_{cenes}	$\ominus$	$\oplus$
Dances with wolves	10000	70	89	17	7	0	0
Avengers1	1804	24	65	12	5	0	0
Avengers2	5341	72	144	20	8	1	0

Table 1: Scene results by the mixed approach

The reported experiments here are done on a limited, but variable, video database. Each video is decomposed of a set of scenes of different lengths. The results shown in table 1 demonstrate the performance of the proposed approach.

5 Comparative analysis and Discussion

For a comparative study the approach is evaluated when the similarity between shots is detected using each descriptor separately without applying the mixed strategy of clustering (see section 3.4). The table 2 summarizes the test results on the first video clip of the Avengers TV movie (*Avengers1*). The parameters (thresholds of spatial/temporal similarities between shots) which have been already determined in an interactive way, when the mixed approach was performed (table 1), are used the same in this experiments.

The measure of similarity between shots is based upon many features. This fact is confirmed by the above experiments. When the construction of scenes is performed using only one descriptor, there is a significant number of missed and false detected scenes. There are many detected scenes decomposed of only one shot. That means the used descriptor for matching is not dominant in these shots. The use of multiple descriptors as proposed by the mixed approach improves the results.

6 Conclusion and perspectives

One challenging problem addressed by the **M**oving **P**icture **E**xpert **G**roup, MPEG-7, is the identification of the scene structure for a movie film [13]. The scene structure allows the end user of the interactive video [7] to access to the video document as to a book (with a table of content). Each scene describes a story or an action of the film.

Descriptor	$\sharp Clu$	$\sharp Scenes$	$\sharp$false $\ominus$	$\sharp$false $\oplus$
histogram	15	7	2	2
auto-correlogram	14	11	0	6
$\sharp$ of similar objects	15	8	0	3

Table 2: Scene results on the *Avengers1* movie film; The scenes are constructed using each descriptor separately.

In this work we have presented a method for identifying the scene structure for real-world movie films. The main part of the proposed method, is the mixed classification of shots. The similarity between shots is based upon multiple descriptors. Two shots are matched on the basis of color-metric histograms, color-metric auto-correlograms and the number of similar objects localized into them. In fact the similarity between two shots is not always a linear combination of different descriptors, but it may be reached using only the dominant descriptor in these compared shots. The mixed classification approach consists in identifying clusters of shots using each descriptor separately (the hierarchical classification algorithm was used). Then, the three obtained partitions of clusters are merged together, based on a distance that measures the degree of overlapping between clusters.

The experiments of the proposed approach on three real-world movie films demonstrate its performance. On the other hand, the reported comparative study confirms the powerful of mixing multiple features as considered by our approach. More expensive experiments on different types of movie films, like comedy, romantic and science fiction films are currently in progress.

Future work will focus the characterization of shots. The matching of objects inter-shots using only the features extracted form their appearances in the key-frames gives poorly results [6]. One direct improvement at this stage is to statistically model the appearances of tracked objects, in the feature space, and then to match them based on these models.

Finally, the drawback of such an approach is the set of parameters which should be chosen carefully for each movie film. For the moment, there is no completely satisfactory method for determining the number of data clusters for the hierarchical classification technique [11] and this point is still a research problem in clustering analysis.

References

[1] J. F. Allen. Maintainning knowledge about temporal intervals. *CACM*, 26:832–843, 1983.

[2] P. Bouthemy and F. Ganansia. Video partitionning and camera motion characterisation for content-based video indexing. *Proc. 3rd IEEE Int. Conf. Image Processing.*, september 1996.

[3] M. Gelgon and P. Bouthemy. A region-level graph labeling approch to motion-based segmentation. In *CVPR*, pages 514–519, Puerto Rico, June 17-19 1997.

[4] R. Hammoud, L. Chen, and F. Fontaine. An extensible spatial-temporal model for semantic video segmentation. In *First Int. Forum on Multimedia and Image Processing, Anchorage, Alaska*, May 1998.

[5] R. Hammoud and R. Mohr. Building and browsing hyper-videos: a content variation modeling solution. *Pattern Analysis and Applications*, 2000. Special Issue on Image Indexation, Submitted.

[6] R. Hammoud and R. Mohr. Gaussian mixture densities for indexing of localized objects in a video sequence. Technical report, INRIA, March 2000. http://www.inria.fr/RRRT/RR-3905.html.

[7] R. Hammoud and R. Mohr. Interactive tools for constructing and browsing structures for movie films. In *ACM Multimedia*, pages 497–498, Los Angeles, California, USA, October 30 - November 3 2000. (demo session).

[8] R. Hammoud and R. Mohr. Probabilistic hierarchical framework for clustering of tracked objects in video streams. In *Irish Machine Vision and Image Processing Conference*, pages 133–140, The Queen's University of Belfast, Northern Ireland, 31 August - 2 September 2000.

[9] J. Huang. *Color Spatial Indexing and Applications.* PhD thesis, Cornell University, August 1998.

[10] M. Irani and P. Anandan. A unified approach to moving object detection in 2d and 3d scenes. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 20(6):577–589, June 1998.

[11] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

[12] J.Aumont and M. Marie. L'analyse des films. *Fac. Cinéma, NATHAN Université*, 1988.

[13] F. Nack and A.T. Lindsay. Everything you wanted to know about mpeg-7. *IEEE Multimedia*, pages 65–77, July-September 1999.

[14] Y. Rui, S. Huang, and S. Mehrota. Exploring video structures beyond the shots. In *Proc. of IEEE conf. Multimedia Computing and Systems*, Austin, Texas USA, June 28-July 1 1998.

[15] Y. Rui, T. S. Huang, and S. Mehrotra. Constructing table-of-content for videos. *ACM Multimedia Systems Journal, Special issue Multimedia Systems on video librairies*, 1999. To appear.

[16] M. Yeung and B. Yeo. Segmentation of video by clustering and graph analysis. *Computer Vision and Image Understanding*, 71(1):94–109, July 1998.

[17] H. J. Zhang, C. Y. Low, S.W. Smoliar, and J.H. Wu. Video parsing, retrieval and browsing: An integrated and content-based solution. *ACM Multimedia*, pages 15–24, 1995.

Chapitre 8

Systèmes interactifs pour la construction et l'utilisation de la vidéo hyperliée

Avoir une bonne idée ne suffit pas !!!
Le maître du jeu, au final, c'est
toujours le client.

Après la présentation des méthodes de fabrication de la structure hiérarchique de la vidéo de bas niveau (i.e. plans, images-clés) et de haut niveau (i.e. groupes d'objets, scènes, ...), ce chapitre a pour objectif de résumer les niveaux d'interactions entre *Homme* (utilisateur), *Machine* et *Vidéo* que nous avons défini et de les intégrer dans nos systèmes interactifs de construction et d'utilisation de la vidéo hyperliée, suite au travail initial conduit par Pascal Bertolino.

8.1 Interactions dans un système de vidéothèque hyperliée

Par analogie avec le standard *MPEG-4* [112] on distingue deux types d'utilisateurs de la vidéo hyperliée: *auteur* et *utilisateur final*. La figure 8.1 illustre les interactions entre Utilisateur/Vidéo, Utilisateur/Machine et Machine/Vidéo dans un système de vidéothèque hyperliée. D'abord, l'utilisateur auteur spécifie un film vidéo à traiter, il fixe les paramètres des méthodes d'indexations automatiques et/ou semi-automatiques à appliquer sur ce film. Ensuite, la machine exécute ces modules et produit une "vidéo hyperliée" qui sera utilisée manuellement par l'utilisateur. A un deuxième niveau d'interaction l'utilisateur final joue la vidéo hyperliée.

La liste suivante résume les tâches de l'auteur de la vidéo hyperliée:

– *Paramétrage* des méthodes d'indexation et de structuration de la vidéo présentées auparavant. A titre d'exemple, lors de la fabrication semi-automatique du niveau

"groupe d'objets" l'auteur désigne les "objets suivis modèles" (voir figure 4.1), ensuite tout autre objet de la séquence vidéo traitée sera classé automatiquement dans l'un d'eux.

– *Édition* des résultats obtenus par les méthodes d'indexation automatique comme nous l'avons précisée dans les chapitres précédents. A titre d'exemple, il corrige le mauvais classement d'un objet suivi par une opération de genre *Drag & Drop* par la souris.

– *Description* textuelle des entités de la vidéo structurée. L'utilisateur auteur intervient pour décrire la sémantique ou l'histoire d'une scène ou d'une classe d'objets du film vidéo.

– *Pointage* d'un objet du film vers un lien extérieur comme l'adresse d'une page web ou l'exécutable d'un programme spécifique (clip de musique, modèle 3D de l'objet d'intérêt, ...).

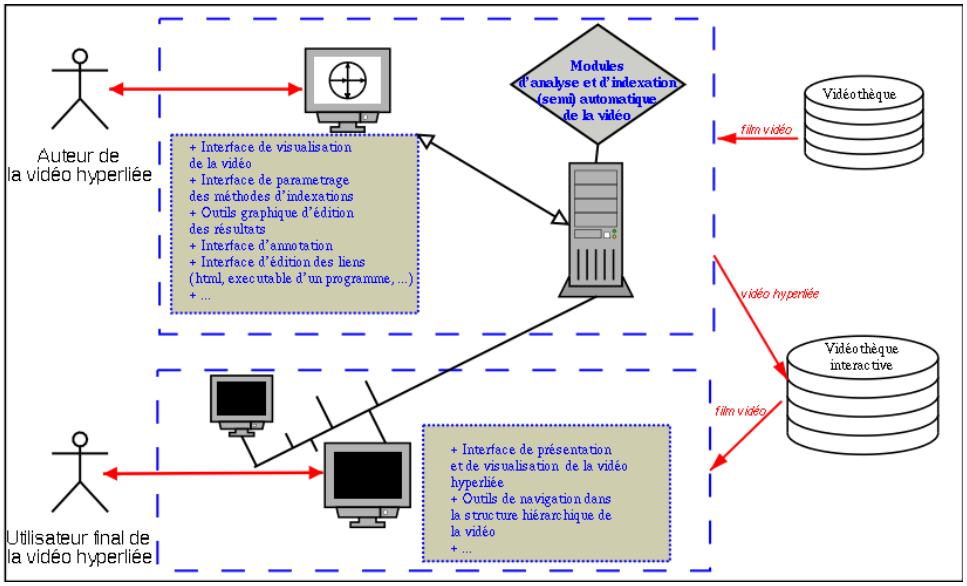


FIG. 8.1: *Nivaux d'interactions entre Utilisateur/Machine/Vidéo dans un système de vidéothèque hyperliée.*

Afin de faciliter les tâches ci-dessus des interfaces d'acquisition des entrées, de visualisation et de présentation des résultats d'indexation ainsi que des outils graphiques simples pour éditer manuellement ces résultats sont indispensables. Aussi, la vidéo hyperliée ainsi construite est présentée à l'utilisateur final via des interfaces graphiques simples et intuitives. C'est le niveau d'interaction Utilisateur/Machine. Les interfaces sont illustrées dans les pages suivantes.

8.2 Résumé de "Interactive Tools for Constructing and Browsing Structures for Movie Films" – ACM'2000

Selon le schéma de la vidéothèque hyperliée que nous venons de présenter, deux interfaces graphiques nommées `VideoPrep` et `VideoClic` ont été développées. La première est conçue pour préparer une vidéo interactive tandis que la seconde est destinée aux utilisateurs finaux pour consulter et naviguer dans la vidéo préparée.

Une première version de ces interfaces a été achevée vers la fin du contrat de collaboration entre Alcatel CRC et l'INRIA (équipes VISTA et MOVI) en septembre 1998, par l'ingénieur expert P. Bertolino. Bertolino a été recruté dans le projet MOVI sur ce contrat [8]. Ma participation à cette version a porté principalement sur l'intégration des modules d'indexation par la couleurs, la conception et le développement de leurs interfaces graphiques. L'indexation des objets est effectuée sur la base des images-médianes représentant les plans comme une solution simple des problèmes abordés dans cette thèse (voir chapitre 3).

Dans le cadre du notre contrat [1] nous avons repris le développement de `VideoPrep` et `VideoClic` en apportant des améliorations pertinentes sur leurs fonctionnements et leurs structures d'indexation, et en intégrant les approches de structuration discutées dans les chapitres précédents de ce mémoire [35]. Une démonstration de la deuxième version de ces systèmes a été présentée à ACM Multimédia [56]. Les détails sont décrits dans les pages suivantes.

Le défaut technique de `VideoPrep` est principalement son traitement des séquences vidéo décompressées en plusieurs formats d'images individuelles (PPM, PGM, GIF) (voir section 2.5). Ceci conduit à l'explosion de l'espace mémoire réquis même pour une séquence vidéo de quelques minutes. Il faudrait repenser à toute l'architecture dans un travail ultérieur; l'utilisation d'un espace disque cache permet vraisemblablement de résoudre ce problème par décompression en temps réel de la bande vidéo MPEG. En ce qui concerne `VideoClic` le téléchargement de toutes les structures de la vidéo hyperliée (plans, classes d'objets, scènes). Pour une large séquence vidéo ceci conduit également à une occupation mémoire trop large. Là encore, un astucieux chargement à la demande permettrait de résoudre ce problème; il faudra juste veiller à ce que ce processus ne ralentisse pas la navigation dans la vidéo.

Malgré leurs défauts techniques nos systèmes sont complets du point de vue recouvrement de la totalité de la structure hiérarchique de la vidéo, automatisation du processus de structuration, souplesse de l'interaction entre l'utilisateur, la vidéo et la machine lors de la création et la présentation de la vidéo hyperliée. Celles-ci les rendent comparables à plusieurs produits concurrents dans le marché comme Mvshots et MoVideo [2] (voir page 13).

1. Contrat sur trois ans entre Alcatel CRC et MOVI démarré le premier Janvier 1998, numéro 198098G
2. http://www.artsvideo.com

Interactive Tools for Constructing and Browsing Structures for Movie Films

Riad Hammoud and Roger Mohr

8th ACM International Conference on Multimedia
November 2000

Abstract: This paper presents a prototype for constructing, browsing and using structures for movie films based on content image analysis only. The goal of the structuring is to facilitate the user's access to the video content (non-linear navigation, etc.). Our prototype provides for the "editor user" advanced tools for structuring the video at its low-level (e.g. shots, key-frames) and high-level structures (e.g. groups of objects, scenes). Also, it provides for the "end user" flexible interfaces for browsing and using constructed video structures.

Against the previous version of this prototype [1], we mainly improved the matching and the clustering of segmented objects. Also, constructing and browsing the high-level scene structure are now available.

1 Constructing structures

Figures 1 and 2 illustrate the system designed for building shots, clusters and scenes structures. It is developed in C++/Ilog-Views and portable under Unix and Linux. In the following, we briefly list its main functionalities. Some modules which implement these functionalities are not completely integrated in the system.

▶ **Basic segmentation.** The partitioning into shots is done firstly using the dominant motion approach. The estimated motion is then used to localize and track mobile objects within shots [2] (figure 1 -bottom). The static objects are manually segmented and tracked.

▶ **Characterizing and matching individual objects.** Individual occurrences of tracked objects are characterized by three different features: global color histograms, color correlograms and local differential invariants. The matching process of individual objects is performed on each descriptor separately. A *linear fusion* of the matching results is adopted when multiple descriptors are used. The weight of each descriptor is fixed by the "editor user" which decides the importance of each descriptor (for example, for this sequence colors are more discriminant than geometric informations).

▶ **Characterizing of tracked objects.** Due to the variable appearance of objects during tracking and the acquisition in poorly constrained dynamic scenes, the matching of individual objects using classical features gives poor results. In order to increase the robustness of existing features, we use the Gaussian mixture densities to model the intra-shot variability of each tracked object [6].

▶ **Clustering of objects.** Both supervised and unsupervised clustering of objects are implemented (figure 2). (1) The user selects by the mouse some tracked objects, considers them as "models" or classes, and then assigns to them all other objects. Currently, this technique is applicable only on modeled tracked objects in the color histogram feature space where the mixture classifier is used to identify classes of individual objects.

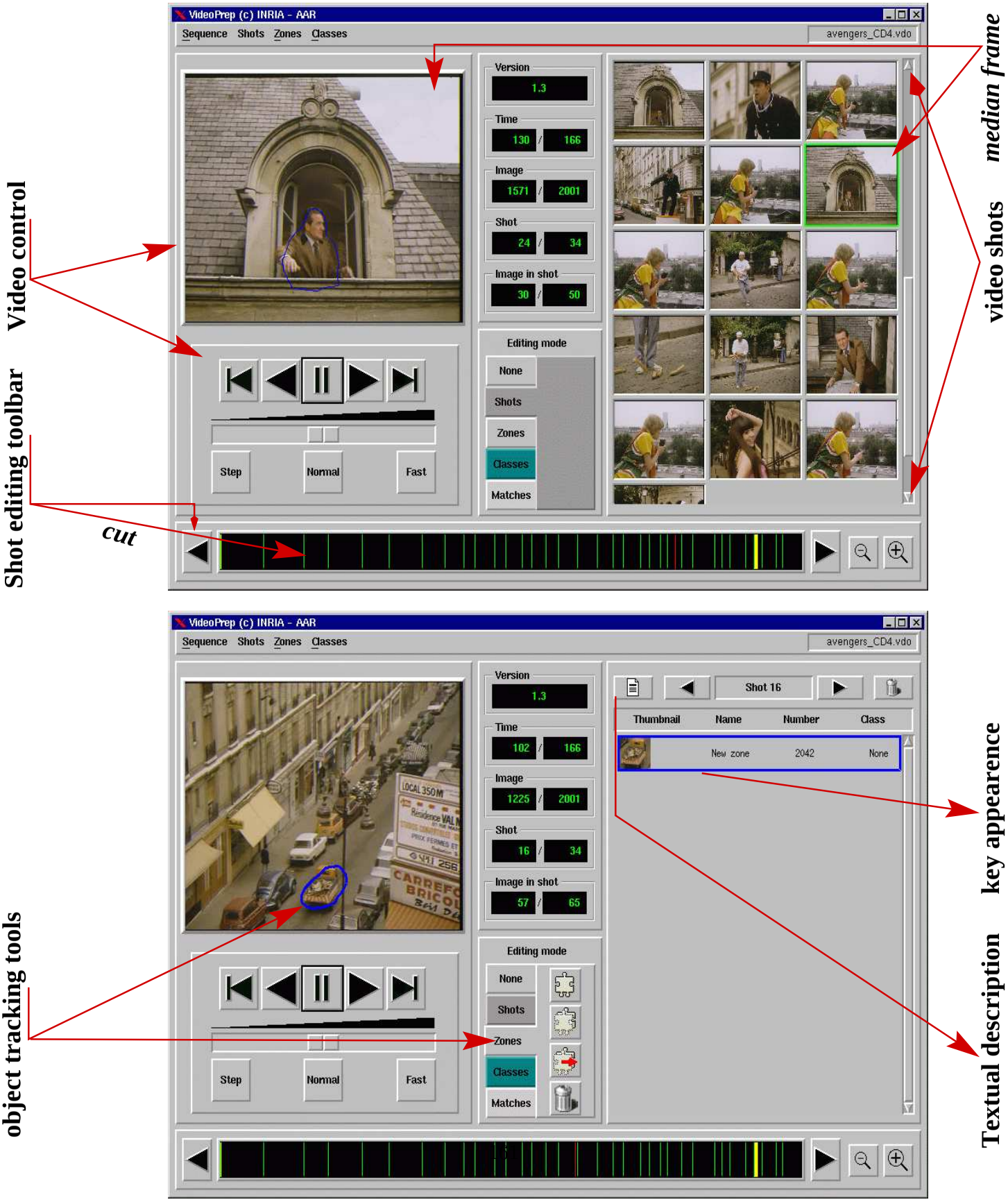


Figure 1: System for constructing video structures : partitioning into shots (top), tracking of objects (bottom)

(2) To avoid a manual selection of "object models", the Ascendant Hierarchical Classification algorithm is used to automatically identify clusters of objects based on different implemented descriptors. The unsupervised classification based on estimated Gaussian mixtures for tracked objects gives good results [5]. The module of this method is not yet integrated in the system.

▶ **User in the loop.** Practically, it is very difficult to perform a perfect clustering of this kind of noisy data (occlusions, illumination changes, etc). The system provides some interactive tools to correct the results of the automatic clustering: (1) *select/browse* clusters at different levels of the hierarchy, (2) *Drag* a badly classified object and *Drop* it into another cluster or a new one.

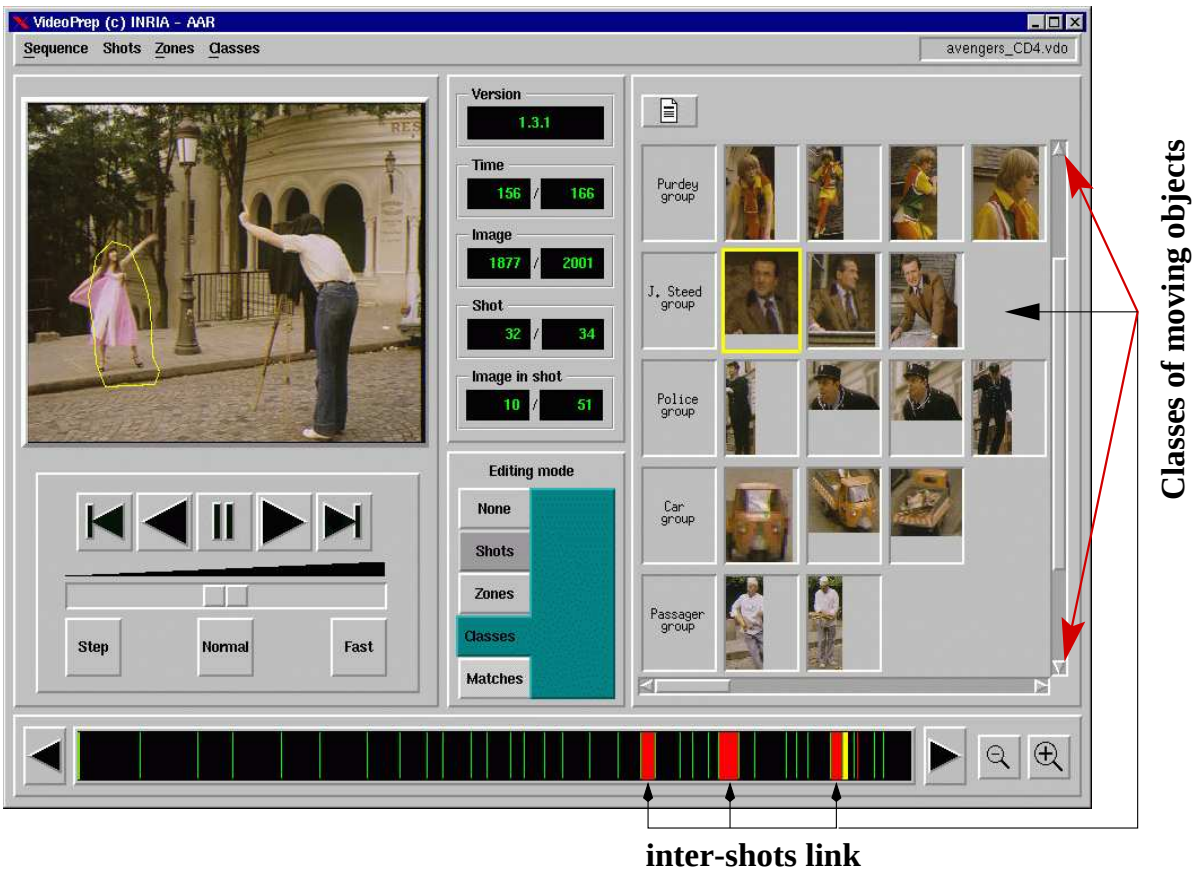


Figure 2: System for constructing video structures : grouping tracked objects into clusters

▶ **key-frames extraction.** A Key-frame is an existing frame which can represent the whole set or a subset of frames of the shot. Usually each shot is represented by only the first frame. In general shots are dynamic, so a single key-frame is not sufficient to represent effectively the content. The modeling of appearances of a tracked object consists in grouping similar views together. An efficient technique is to select from each group of similar views the median image as a key-frame (see [5]).

▶ **Scenes extraction.** A video scene is defined as a collection of semantically related and temporally adjacent shots, depicting and conveying a high-level concept or story. Our approach to extract scenes is an extension of the method of [3]. The method consists firstly in grouping similar shots, of the same predefined temporal window and the same "narrative sequence", into clusters, then exploring the temporal graph of clusters to extract scenes. The temporal relations of Allen (meets, before, ...) are used to connect the nodes of the graph. A scene is formed by merging nodes (clusters) of a sub-graph, which does not contain a temporal relation of type "meets" that can disconnect it into two other sub-graphs. The extension of this method is done at the clustering stage. Three descriptors are used to match similar shots represented by key-frames: histograms, correlograms and the number of similar objects in two compared shots. On each descriptor the hierarchical classification algorithm is performed where the number of clusters is determined using a predefined threshold. Each one of these descriptors summarizes differently the content of a shot. So, the obtained clusters by different descriptors are not necessarily similar. A distance that measures the intersection between two clusters of two different descriptors is computed. Here the goal is to deduce from the three sets of clusters only one set. Based on this, we construct the temporal graph of clusters from which the scenes are extracted as explained previously. The experimental results depict in [4] shown the performance of this extended method against its original form. The related modules to this functionality are in the course of being integrated into this system.

2 Browsing and using structures

Once the constructing structures for a movie film is achieved, the "end-user" has the ability to explore the content of the film in a new way. The cluster structure defines in the movie film links between objects. At this level, the end-user clicks an object of interest (actor, car, ...), jumps to its next or previous occurrence in the film, plays the corresponding shot, plays the corresponding action, discovers a related WWW link, etc. The scene structure allows the end user to access the video document as a book (with a table of contents). Each scene describes a story or action of the film.

Figure 3 illustrates the end-user interface for browsing and navigation in the different structure level of the movie film. This interface is developed in `Java` in order to be used on different user platforms. The next version of this application will be accessible on the World Wide Web.

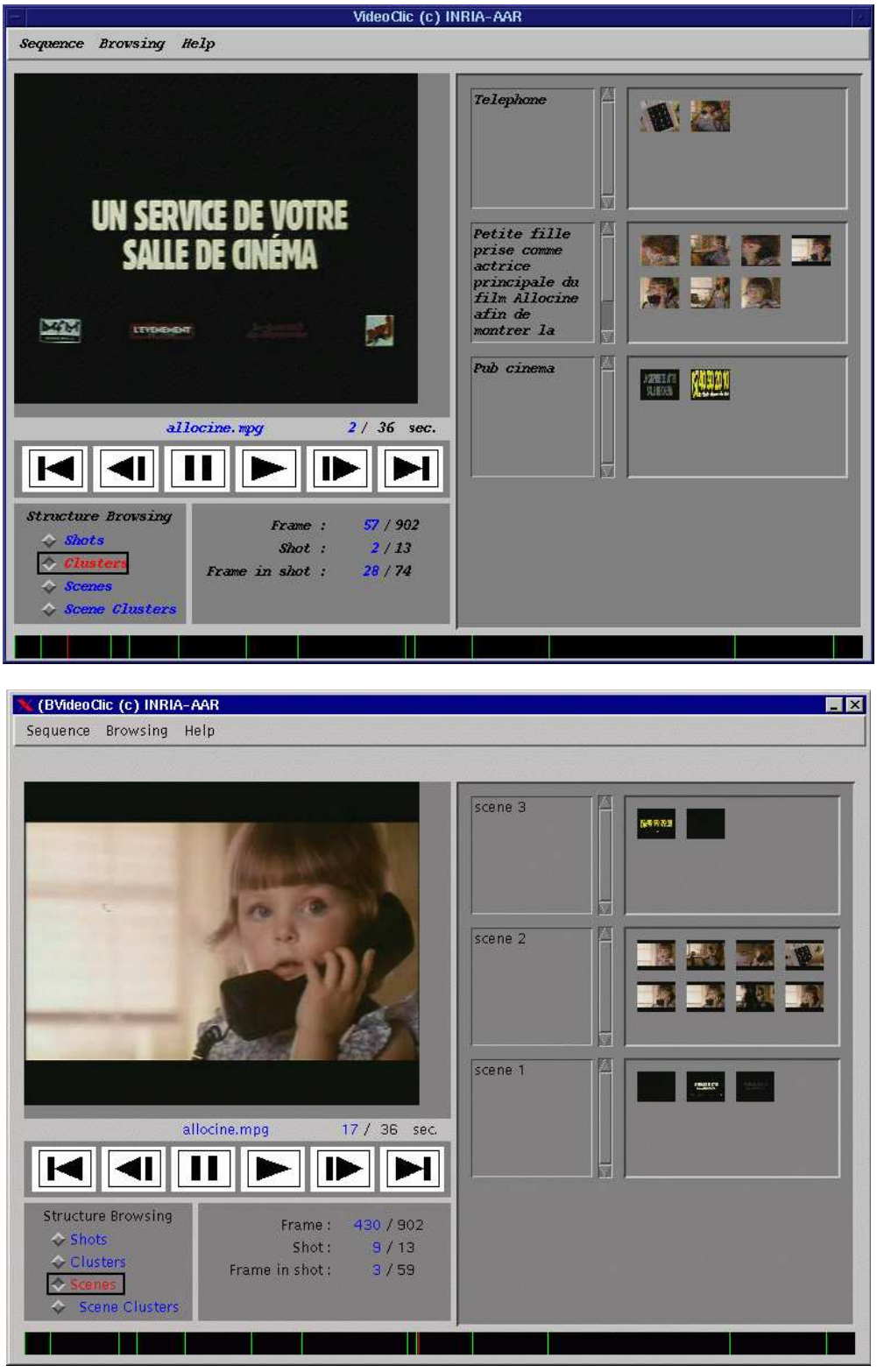


Figure 3: End-user system for browsing video structures : (top) browsing clusters, (bottom) browsing scenes.

References

[1] S. Benayoun, H. Bernard, P. Bertolino, M. Gelgon, C. Schmid, and F. Spindler. Structuration de vidéos pour des interfaces de consultation avancées. In *CORESA 98 – Journées d'études et d'échanges COmpression et REprésentation des Signaux Audiovisuels.*, June 1998.

[2] M. Gelgon and P. Bouthemy. A region-level graph labeling approch to motion-based segmentation. In *CVPR*, pages 514–519, Puerto Rico, June 17-19 1997.

[3] R. Hammoud, L. Chen, and F. Fontaine. An extensible spatial-temporal model for semantic video segmentation. In *First Int. Forum on Multimedia and Image Processing, Anchorage, Alaska*, 1998.

[4] R. Hammoud and D. G. Kouam. A mixed classification approach of shots for constructing scene structure for movie films. In *Irish Machine Vision and Image Precessing Conference*, pages 223–230, The Queen's University of Belfast, Northern Ireland, 31 August-2 Septembre 2000.

[5] R. Hammoud and R. Mohr. Building and browsing hyper-videos: a content variation modeling solution. *Pattern Analysis and Applications*, 2000. Special Issue on Image Indexation, Submitted.

[6] R. Hammoud and R. Mohr. Mixture densities for video objects recognition. In *International Conference on Pattern Recognition*, volume 2, pages 71–75, Barcelona, Spain, 3-8 September 2000.

Chapitre 9

Conclusion générale

Saluer ce qui va pour mieux dire
ce qui ne va pas.

Dans le contexte du prochain standard proclamé MPEG-7, nous avons présenté un travail qui touche différents problèmes liés à l'indexation de la vidéo par le contenu. Nos contributions fondamentales ont porté sur la caractérisation des objets suivis par des modèles statistiques adaptés à leurs changements d'apparences intra-plan, la classification automatique et semi-automatique des objets suivis basée sur les modèles estimés, la représentation optimale d'un objet suivi (apparences-clés), et enfin sur la classification des plans adjacents en des groupes sémantiques (macro-segmentation en scènes). L'objectif de telles tâches automatiques est d'optimiser les coûts d'interaction entre l'utilisateur et la machine lors de la construction et la présentation d'un document vidéo afin de satisfaire une partie des besoins de MPEG-7 ou de permettre une interactivité Homme/Vidéo.

9.1 Bilan de travail

Le problème principal à résoudre pour associer des entités dans une vidéo est la grande variabilité de leurs apparences. Cette variabilité intra-plan est due principalement aux changements d'apparence de l'objet au cours du temps et au manque de robustesse des descripteurs classiques à ces changements de l'image, sans compter les occultations qui ne sont pas réellement modélisables dans ce contexte.

Nous nous sommes penchés d'abord sur ce problème de la variabilité intra-plan d'un objet suivi dans l'espace de descripteurs de couleurs. Ceci conduit à une représentation spatiale dans l'espace de descripteurs qui n'est pas compacte et souvent multimodale. Pour modéliser cet ensemble d'aspects, nous avons retenu un modèle statistique : une distribution représentée par un mélange de lois gaussiennes. L'estimation de cette distribution est vue comme un problème de recherche des classes d'apparences intra-plan d'un objet suivi. Le mélange gaussien multivariée dans le cas automatique possède des propriétés

intéressantes : on peut montrer qu'il peut bien approcher des distributions complexes ; de plus il existe maintenant des outils qui permettent de sélectionner automatiquement le modèle gaussien le mieux adapté aux données et le nombre convenable de classes d'apparences intra-plan, structure qui diffère d'un objet suivi à un autre.

Les expérimentations de cette approche montrent que les distributions des objets suivis correspondent bien à des mélanges de loi gaussiennes, et que le choix automatique de la structure du mélange est bien adapté aux données mais par contre ce choix n'est pas toujours optimal.

L'intérêt de cette approche est double : (1) La caractérisation des objets suivis par des modèles de mélange gaussien permet de faire une mise en correspondance efficace entre eux qui tient compte des changements intra-plan d'apparence des objets. (2) L'indexation des objets suivis d'une large séquence vidéo sur la base des paramètres gaussiens est maintenant faisable vu la taille raisonnable de ces descripteurs statistiques et le nombre limité des classes d'apparences retenues pour une séquence vidéo.

Cette approche de modélisation de la variabilité nous a permis d'aborder ensuite le problème de classification des objets suivis. La classification des apparences d'objets suivis à travers la vidéo dans des groupes homogènes créee des liens entre les objets "identiques" du film. Ces liens peuvent être explorés par l'utilisateur final de la vidéo hyperliée.

Dans un premier temps, nous avons proposé une approche de classification semi automatique des objets suivis. L'auteur de la vidéo hyperliée choisit d'une manière interactive la liste des modèles d'objets suivis pour une séquence vidéo. Les mélanges gaussiens estimés pour ces modèles d'objets sont liés ensuite par une loi globale de mélange gaussien. Nous avons proposé d'utiliser le maximum a posteriori (MAP) pour classer toute apparence dans le film dans l'un des modèles d'objets. En intégrant l'aspect temporel une deuxième fois lors du classement des apparences, nous avons proposé d'utiliser une méthode robuste, le vote majoritaire, pour classer toutes les apparences d'un objet suivi dans le même modèle d'objet suivi.

L'étude comparative menée dans ce chapitre a montré une amélioration considérable allant jusqu'au 35% des pourcentages de bon classement des apparences d'objets suivis par l'approche proposée vis-à-vis de la méthode classique de "moyenne temporelle de descripteurs". La méthode de classement robuste donne des résultats meilleurs de l'ordre de 10% que la méthode de MAP. La séquence vidéo Avengers-1 utilisée dans ces expériences contient des modèles d'objets suivis de nombre d'apparences très limités. Cette situation explique la dégradation des résultats lorsque le nombre de composantes gaussiennes dépasse 2. On s'aperçoit dans ce cas que les critères BIC et ICL ont tendance à choisir un nombre de classes d'apparences élevé mais avec des modèles gaussiens linéaires. Les mauvais résultats de classement sont obtenus quand le critère d'entropie NEC est employé.

La performance de l'approche est la meilleure quand la modélisation est effectuée dans l'espace réduit des histogrammes de couleurs RGB. Par contre elle est la plus mauvaise dans l'espace réduit des histogrammes de couleurs normalisés rgb. La normalisation des couleurs conduit théoriquement à une représentation invariante aux changements de luminosité mais elle conduit aussi à une réduction drastique de l'information; il en est de même pour la réduction de l'espace des histogrammes normalisés par une analyse en composantes principales; tout cela explique la dégradation des résultats, en particulier dans ces cas on

observe un chevauchement important entre les modèles gaussiens des différents modèles d'objets suivis.

Dans un second temps, nous avons proposé une automatisation de l'approche de classification des objets suivis. La classification ascendante hiérarchique est appliquée sur la matrice de distances entre les objets suivis. La distance proposée entre deux objets suivis est la distance minimale de Kullback (ou de Bhattacharyya) entre leurs classes d'apparences intra-plan. Des résultats satisfaisants de l'ordre de 80% ont été obtenus dans ce cas automatique et sur une base d'objets vidéo très complexe. Il ressort de cette validation expérimentale que les résultats sont les meilleurs lorsque la distance de Kullback entre deux classes d'apparences et la distance de "saut maximum" entre deux groupes d'objets suivis dans la hiérarchie construite sont employées. Une sélection interactive du nombre de classes d'objets ainsi que des outils interactifs pour éditer les résultats de la classification ont été proposés dans un système de vidéothèque hyperliée.

Nous nous intéressons ensuite au problème de l'extraction automatique des apparences-clés d'un objet suivi. L'extraction des apparences-clés permet une visualisation rapide du contenu d'un plan et un appariement de deux objets suivis (deux plans) sur la base des apparences-clés. Pour résoudre ce problème, nous proposons une technique de classification automatique, le mélange gaussien, pour regrouper les apparences similaires dans des classes d'apparences intra-plan. Ensuite, nous effectuons une sélection des apparences médianes de ces classes. Et enfin nous ne gardons que les apparences-clés les plus représentatives qui vérifient deux tests de dissimilarité spatiale et temporelle. L'avantage de cette approche est qu'elle permet une sélection automatique et optimale des apparences les plus représentatives de l'objet suivi. Les expérimentations ont montré l'efficacité et la facilité de la mise en oeuvre d'une telle approche.

Afin de compléter la construction automatique de la structure de haut niveau d'un film vidéo, nous avons abordé à la fin de cette thèse le problème d'identification des scènes. Ce problème est très délicat parce que d'une part il n'existe pas une formulation précise de la rupture entre deux scènes et d'autre part il est lié aux problèmes de caractérisation fiable du contenu des plans et de classification automatique des plans. Nous avons étendu un travail antérieur sur la macro-segmentation en proposant d'apparier deux plans vidéo sur la base de plusieurs descripteurs de bas niveau: le contenu du plan, la distribution globale des couleurs et la distribution spatiale des couleurs extraites de l'image médiane du plan. La distance proposée pour mesurer la similarité entre deux plans sur la base du descripteur du contenu est le nombre des objets similaires de deux plans. Il ressort à ce point un problème difficile de fusion de descripteurs hétérogènes à résoudre. Nous avons proposé un algorithme de fusion des groupes de plans des différents descripteurs plutôt que les descripteurs eux mêmes. Une amélioration nette des résultats de l'approche de base est réalisée sur plusieurs séquences vidéo. Lors de la construction des groupes de plans selon un descripteur donné, un seuil de dissimilarité spatiale et un autre de voisinage temporel sont fixés a priori. Il est normal que les résultats dépendent de ces deux paramètres de seuillage qui ne sont pas facile à choisir dans certains cas.

Une fois que nous avons défini les structures de bas et de haut niveau d'un film vidéo, nous avons développé deux prototypes (VideoPrep et VideoClic) qui facilitent les interactions Homme/Machine et Machine/Vidéo, lors de la construction et l'utilisation d'une

vidéo interactive.

9.2 Perspectives de recherche

Les points suivants peuvent être améliorés d'une façon immédiate :

Classification robuste des apparences intra-plan Lors du suivi d'un objet à travers le temps une certaine continuité dans l'apparence de l'objet est généralement conservée. Or quelques-unes de ces apparences sont parfois extrêmes. Celles-ci résultent par exemple des occultations partielles significatives (voir les dernières apparences de la figure 3.2). La représentation de ces apparences dans l'espace de descripteurs correspond à des points atypiques (*outliers*). Lors de l'étape de la modélisation de la variabilité intra-plan, on constate que de tels points tendent à former des classes d'apparences comportant un faible nombre de représentants ou bien des classes d'apparences avec des variances très grandes. L'algorithme EM réestime les centres des classes à chaque itération. Il faut donc s'attendre à ce que cet algorithme se montre assez sensible à la présence de valeurs atypiques. Il serait donc très intéressant de rendre le processus moins sensible aux valeurs extrêmes. Les algorithmes robustes de k-médianes ou de Ransac pourraient être envisageables pour ce problème.

Apprentissage mixte Dans le cas de l'approche de classification supervisée, une relaxation de la désignation d'un objet suivi modèle sur plusieurs plans vidéo (par exemple les vues de la voiture Mercedes du haut -filmé du ciel- et de face) permettrait de résoudre le problème des données limitées, de prendre en compte dans la modélisation statistique de l'apparence intra-plan plusieurs aspects de l'objet suivi, et donc d'augmenter le taux de reconnaissance des apparences requêtes.

Appariement mixte L'identification de deux plans ne peut pas être détectée sur la base d'un seul descripteur comme nous l'avons vu dans le chapitre 7. D'autre part l'étude présentée dans le chapitre 3 montre que l'identification de deux objets sur la base de leurs deux apparences médianes n'est pas la bonne stratégie. En représentant le plan vidéo par un nombre optimal d'images-clés (extension de l'approche de sélection des apparences-clés) nous envisageons d'effectuer un appariement mixte (plusieurs images-clés et plusieurs descripteurs) de deux plans. La difficulté ici est la définition d'une fonction statistique qui tranche entre les deux hypothèses de ressemblance ou dissimilarité.

Les aspects suivants demandent un travail plus important :

Modélisation des régions d'objets suivis La modélisation des objets suivis dans ce travail est faite par des descripteurs globaux. Pour rendre compte des parties qui apparaissent et disparaissent il serait intéressant d'avoir des descripteurs plus locaux. Ces descripteurs peuvent être centrés sur des points d'intérêts (voir section 3.3) ou des régions.

Entre ces descripteurs locaux existent des relations de position comme "voisin et au-dessus". Il faut dans ce cas modéliser chaque descripteur local et pour cela les algorithmes EM peuvent à nouveau être utilisés, mais une telle approche nécessite le suivi de ces descripteurs, et, pour ne pas perdre d'information, la prise en compte de contraintes de positionnement relatif.

Relevance feedback Revenons sur le cas d'une classification automatique par le CAH appliqué à une matrice de distances entre les classes d'apparences intra-plan des objets. Nous avons observé expérimentalement qu'un retour par l'utilisateur d'une version retouchée "des classes d'objets" à un niveau très fin de la hiérarchie permet probablement de poursuivre automatiquement une construction correcte des classes d'objets à des niveaux plus élevés dans la hiérarchie. Une intégration de l'approche "relevance feedback" dans la construction des groupes d'objets est une voie intéressante. Reste à savoir comment et à quel niveau l'appliquer ?

Vers une modélisation intra-plan généralisée La densité du mélange gaussien est couramment utilisée en statistique pour modéliser des distributions gaussiennes. Mais il est très difficile de vérifier cette hypothèse de normalité dans des espaces de descripteurs à grandes dimensions. On pourrait considérer l'utilisation d'une loi générale comme la loi statistique gamma qui permet de modéliser n'importe quel type de distribution. Mais il faut être capable d'estimer les paramètres du mélange de lois gammas et mettre en oeuvre des critères de sélection de la structure du mélange.

Vers une sélection automatique des descripteurs Le nombre de descripteurs calculables sur les images est très grand. Le problème est de savoir comment déterminer automatiquement quel est le descripteur le plus discriminant pour une image donnée. Les statistiques doivent apporter une réponse, mais il faut garder à l'esprit que le problème est difficile car on ne dispose que de quelques images exemples, de beaucoup de contre-exemples, et que le nombre de descripteurs envisageables est infini.

Références bibliographiques

[1] P. Aigrain, P. Joly, and V. Lougueville. Medium knowledge-based macro-segmentation of. In *Proc. IJACI workshop on intelligent multimedia information retreival*, ED. Mark Maybury, Montreal, 1995.

[2] H. Akaike. A new look at the statistical model identification. IEEE *Transactions on Automation and Control*, AC-19(6):716–723, December 1974.

[3] S. Antani, R. Kasturi, and R. Jain. Pattern recognition methods in image and video databases: Past, present and future. In *Advances in Pattern Recognition Joint IAPR International workshops*, Sydney, Australia,, August 1998.

[4] M. Ardebilian, X. W. TU, L. Chen, and P. Faudemay. Video segmentation using 3d hints contained in 2d images. *SPIE*, 2916, 1996.

[5] Y. S. Avrithis, A. D. Doulamis, N. D. Doulamis, and S. D. Kollias. A stochastic framework for optimal key frame extraction from mpeg video databases. *Computer Vision and Image Understanding*, 75(1/2):3–24, July-August 1999.

[6] J.D. Banfield and A.E. Raftery. Model-based Gaussian and non-Gaussian Clustering. *Biometrics*, 49:803–821, 1993.

[7] S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color-and texture-based image segmentation using EM and its application to content-based image retrieval. In *Proceedings of the 6th International Conference on Computer Vision, Bombay, India*, pages 675–682, January 1998.

[8] S. Benayoun, H. Bernard, P. Bertolino, M. Gelgon, C. Schmid, and F. Spindler. Structuration de vidéos pour des interfaces de consultation avancées. In *CORESA 98 – Journées d'études et d'échanges COmpression et REprésentation des Signaux Audiovisuels.*, June 1998.

[9] C. Biernacki. *Choix de Modèles en Classification.* PhD thesis, Université de technologie de Compiègne, septembre 1997.

[10] C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the integrated classification likelihood. Rapport de recherche RR-3521, INRIA, October 1998.

[11] G. Bisson. Définition de la notion de similarit dans les modles objets. *LMO'94*, pages 53–68, 1994.

[12] R.M. Bolle, B.-L.Yeo, and M.M.Yeung. Video query: Research directions. *IBM Journal of Research and Development*, 42(2), 1998.

[13] J. S. Boreczky and L. A. Rowe. Comparison of video shot boundary detection techniques. In *Proc. SPIE Conf. on Vis. Commun and Image Proc.*, 1996.

[14] P. Bouthemy and F. Ganansia. Video partitionning and camera motion characterisation for content-based video indexing. *Proc. 3rd IEEE Int. Conf. Image Processing.*, september 1996.

[15] H. Bozdogan. *Multi-Sample Cluster Analysis and Approaches to Validity Studies in Clustering Individuals.* Thèse de doctorat d'etat, University of Illinois, Chicago, 1981.

[16] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Santa Barbara, California, USA*, pages 8–15. IEEE Computer Society Press, June 1998.

[17] I.V. Cadez, C.E. McLaren, P. Smyth, and G. J. McLachlan. Hierarchical models for screening of iron deficiency anemia. In *The Sixteenth International Conference on Machine Learning*, Bled, Slovaine, 27-30 June 1999.

[18] L. Cammoun. Variabilité et performance des invariants locaux sur une base d'objets vidéo, Mars 2000. rapport de stage effectué au sein de l'équipe MOVI dans le cadre d'un projet de coopération entre l'ENSI de Tunise et l'INRIA. Responsables: Riad Hammoud et Roger Mohr.

[19] J. Canny. A computational approach to edge detection. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[20] C. Carson, M. Thomas, S. Belongie, J. Hellerstein, and J. Malik. Object retrieval - blobworld: A system for region-based image indexing and retrieval. *Lecture Notes in Computer Science*, 1614:509–516, 1999.

[21] G. Celeux and G. Govaert. A Classification EM Algorithm for Clustering and Two Stochastic Versions. *Computational Statistics & Data Analysis*, 14:315–332, 1992.

[22] G. Celeux and G. Govaert. Gaussian Parsimonious Models. *Pattern Recognition*, 28(5):781–783, 1995.

[23] G. Celeux and G. Soromenho. An Entropy Criterion for Assessing the Number of Clusters in a Mixture Model. *Journal of Classification*, 13(2):195–212, 1996.

[24] S.F. Chang and D.G. Messerschmitt. Manipulation and compositing of mc-dct compressed video. *IEEE Journal on Selected Areas of Communications*, 13(1):1–11, 1995.

[25] L. Chen, D. Fontaine, and R. Hammoud. La segmentation de la vidéo basée sur les indices spatio-temporelles. In *Actes des 4émes Journées d'Etudes et d'Echanges Compression et Représentation des Signaux Audiovisiuels (CORESA'98)*, CNET-Lanion, France, 9-10 Juin 1998.

[26] F. S. Cohen, Z. Huang, and Z. Yang. Invariant matching and identification of curve using b-splines curve representation. IEEE *Transactions on Image Processing*, 4(1):1–10, January 1995.

[27] R. Conners and C. Harlow. Equal probability quantizing and texture analysis of radiographic images. *Computer Graphics and Image Processing*, 8:447–463, 1978.

[28] A. Cutler and M. Windham. Information-based validity functionals for mixture analysis. In *First US-Japan Conference on the Frontiers of Statistical modeling*, pages 149–170, Amsterdam, 1993.

[29] J. Davis and A. F. Bobick. The representation and recognition of human mouvement using temporal templates. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Puerto Rico, USA*, 1997.

[30] P. Demartines and J. Herault. Curvilinear component analysis: A self–organizing neural network for nonlinear mapping of data sets. IEEE *Transactions on Neural Networks*, 8(1):148–154, 1997.

[31] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data (with discussion). *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[32] E. Diday and J.C. Simon. Clustering analysis. In K.S. Fu, editor, *Communication and Cybernetics*. Springer-Verlag, 1976.

[33] N. Druga. Indexation par des indices d'images. Rapport de DEA IVR, ENSIMAG, June 1998.

[34] B. Dubuisson. *Diagnostic et reconnaissance de formes*. Hermès, Paris, France, 1990.

[35] G. Durand and R. Hammoud. Videoprep: Nouvelles fonctionalités et améliorations. Technical report, Convention Alcatel AAR/INRIA, June 1999.

[36] G.D. Finlayson and G.Y. Tian. Color normalization for color object recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 13(8):1271–1285, 1999.

[37] D. Forsey and D. Wong. Surface fitting with hierarchical splines. *ACM Transactions on Graphics*, 14(2):134–161, April 1995.

[38] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, 1990.

[39] B.V. Funt and G.D. Finlayson. Color constant color indexing. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 17(5):522–529, 1995.

[40] U. Gargi, R. Kasturi, and S. Antani. Performance characterization and comparison of video indexing algorithms. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR'98)*, pages 559–565, Santa Barbara, Juin 1998.

[41] M. Gelgon. *Segmentation spatio-temporelle et suivi dans une séquence d'images : application à la structuration et à l'indexation de vidéo.* Thèse de doctorat d'etat, Université de Rennes I, Rennes, France, 1998.

[42] M. Gelgon and P. Bouthemy. A region-level graph labeling approch to motion-based segmentation. In *CVPR*, pages 514–519, Puerto Rico, June 17-19 1997.

[43] T. Gevers and A. W. M. Smeulders. Image indexing using composite color and shape invariant features. In *Proceedings of the 6th International Conference on Computer Vision, Bombay, India*, pages 576–581, January 1998.

[44] T. Gevers and A.W.M. Smeulders. Color-metric pattern-card matchning for viewpoint invariant image retrieval. In *Proceedings of the 13th International Conference on Pattern Recognition, Vienna, Austria*, 1996. see http://zomax.wins.uva.nl:5345/zomax/HTML/pub.html.

[45] T. Gevers and A.W.M. Smeulders. Image indexing using composite color and shape invariant features. In *Proceedings of the 6th International Conference on Computer Vision, Bombay, India*, pages 576–581, 1998.

[46] T. Gevers and H. Stokman. Reflectance based edge classificatin. In *Proceedings of the 12th Conference on Vision Interface, Trois Rivières. Québec*, pages 25–32, 1999.

[47] C. Goble, M. O'Docherty, P. Crowther, M. Ireton, J. Oakley, and C. Xydeas. The manchester multimedia information system. In *Proc E. D. B. T.'92 Conf. on Advances in Database Technology*, volume 580, pages 39–55, 1994.

[48] Y. Gong. *Intelligent Image Databases: towards Advances Image Retrieval.* Academic Publishers, 1998.

[49] P. Gros, G. Mclean, R. Delon, R. Mohr, C. Schmid, and G. Mistler. Utilisation de la couleur pour l'appariement et l'indexation d'images. Technical Report 3269, INRIA, September 1997.

[50] N. Guimaraes, N. Correia, I. Oliveira, and J. Martins. Designing computer for content analysis: A situated use of video parsing and analysis techniques. *Multimedia Tools and Applications*, 7:159–180, 1998.

[51] R. Hammoud. La segmentation en sècnes de la vidéo basée sur les indices spatio-temporels. Mémoire de dea, Laboratoire HEUDIASYC de l'Universit de technologie de Compigne, Septembre 1997.

[52] R. Hammoud and L. Chen. A spatiotemporal approach for semantic video macro-segmentation. In *European Workshop on Content-Based Multimedia Indexing*, pages 195–201, IRIT-Toulouse FRANCE, Octobre 1999.

[53] R. Hammoud, L. Chen, and F. Fontaine. An extensible spatial-temporal model for semantic video segmentation. In *First Int. Forum on Multimedia and Image Processing, Anchorage, Alaska*, May 1998.

[54] R. Hammoud and D. G. Kouam. A mixed classification approach of shots for constructing scene structure for movie films. In *Irish Machine Vision and Image Precessing Conference*, pages 223–230, The Queen's University of Belfast, Northern Ireland, 31 August-2 Septembre 2000.

[55] R. Hammoud and R. Mohr. Gaussian mixture densities for indexing of localized objects in a video sequence. Technical report, INRIA, March 2000. http://www.inria.fr/RRRT/RR-3905.html.

[56] R. Hammoud and R. Mohr. Interactive tools for constructing and browsing structures for movie films. In *ACM Multimedia*, pages 497–498, Los Angeles, California, USA, October 30 - November 3 2000. (demo session).

[57] R. Hammoud and R. Mohr. Mixture densities for video objects recognition. In *International Conference on Pattern Recognition*, volume 2, pages 71–75, Barcelona, Spain, 3-8 September 2000.

[58] R. Hammoud and R. Mohr. A probabilistic framework of selecting effective key frames for video browsing and indexing. In *International workshop on Real-Time Image Sequence Analysis*, pages 79–88, Oulu, Finland, Aug. 31-Sep. 1 2000.

[59] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, pages 147–151, 1988.

[60] R.J. Hathaway. Another Interpretation of the EM Algorithm for Mixtures Distributions. *Statistics and Probability Letters*, 4:53–56, 1986.

[61] P.V.C. Hough. A method and means for recognition complex patterns. *U.S. Patent*, 1962.

[62] J. Huang. *Color-spatial image indexing and applications.* PhD thesis, Cornell university, August 1998.

[63] J. Huang, S. Ravi Kumar, M. Mitra, W.J. Zhu, and R. Zabih. Image indexing using color correlograms. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Puerto Rico, USA*, pages 762–768, June 1997.

[64] I.Haritaoglu, D.Harwood, and L.Davis. A real time system for detecting and tracking people. *Accepted for journal publication Image and Vision Computing Journal*, January 1999.

[65] INA. *L'INA, avec le dépôt légal de la radio-télévision (inathèque), s'installe à la bibiolthèque nationale de France.*, Octobre 1998. http://www.ina.fr/Actualite/Communiques/1998/10/01.0.fr.html.

[66] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, 1988.

[67] A.K. Jain and A. Vailaya. Image retrieval using color and shape. *Pattern Recognition*, 29:1233–1244, 1996.

[68] J.Aumont and M. Marie. L'analyse des films. *Fac. Cinéma, NATHAN Université*, 1988.

[69] S.C. Johnson. Hierarchical clustering schemes. *Psychometrika*, pages 241–254, 1967.

[70] P. Joly. *Consultation et Analyse des Documents en Image Anime Numrique.* PhD thesis, Thèse de l'Université Paul Sabatier de Toulouse, July 1996.

[71] R.E. Kass and A.E. Raftery. Bayes factors. *Journal of American Statistics Association*, 90(430):773–795, June 1995.

[72] R. Kasturi and R. Jain. Dynamic vision. *IEEE Computer Soc.*, 1990.

[73] H. Klock and J. Buhmann. Multidimensional scaling by deterministic annealing. In *International workshop on energy minimisation methods in computer vision and pattern recognition*, pages 245–260, Venice, 1997.

[74] S. Kullback. *Information theory and Statistics.* Dover, New York, NY, 1968.

[75] L. Lebart, A. Morineau, and M. Piron. *Statistique exploratoire multidimensionnelle.* Dunod, 1995.

[76] R. Lienhart, W. Effelsberg, and R. Jain. Visualgrep: A systematic method to compare and retrieve video sequences. In *In Proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases VI*, volume SPIE 3312, pages 271–282, 1997.

[77] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.

[78] W.Y. Ma and B.S. Manjunath. Texture features and learning similarity. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, pages 425–430, 1996.

[79] L. MacQuitty. Similarity analysis by reciprocal pairs of discrete and continuous data. *Educ. Psych. Meas*, 26:825–831, 1996.

[80] W. Mahdi, M. ARDABILIAN, and L. Chen. Automatic video scene segmentation based on spatial-temporal clues and rhythm. *International Journal of Networking and Information Systems*, 5, January 2000.

[81] W. Mahdi, L. Chen, and D. Fontaine. Improving the spatial-temporal clue based segmentation by the use of rhythm. In *2nd European Conference on research and advanced Technology for Digital Libraries (ECDL98)*, pages 169–182, Heraklion, Crete, Greece, Septembre 1998.

[82] S.J. Mckenna, S. Gong, and Y. Raja. Modelling facial colour and identity with gaussian mixtures. *Pattern Recognition*, 31(12):1883–1892, 1998.

[83] G.J. Mclachlan and K.E. Basford. *Mixture, Models, Inference and Applications to Clustering*. Marcel Dekker, New York, 1988.

[84] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[85] J. Monaco. *How to read a film - The Art, Technology, Language, History and Theory of Film and Media*, 1981.

[86] J.L. Mundy and A. Zisserman. Introduction - towards a new framework for vision. In J.L. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pages 1–39. The MIT Press, Cambridge, MA, USA, 1992.

[87] F. Nack and A.T. Lindsay. Everything you wanted to know about mpeg-7. *IEEE Multimedia*, pages 65–77, July-September 1999.

[88] M. Nadler and E. Smith. *Pattern Recognition Engineering*. Wiley Interscience, John Wiley and sons inc., 1992.

[89] A. Nagazaka and Y. Tanaka. Automatic video indexing anf full-video search for objects appearences. *Visual Database Systems II*, pages 113–127, 1992.

[90] C. Nastar, N. Boujemaa, M. Mitschke, and C. Meilhac. Surfimage: Un système flexible d'indexation et de recherche. In *CORESA 98 – Journées d'études et d'échanges COmpression et REprésentation des Signaux Audiovisuels.*, Lannion, France, June 1998.

[91] W. Niblack, X. Zhu, J. L. Hafner, T. Breuel, and et al. Updates to the qbic system. In *In proceedings of SPIE Conference on Storage and Retrieval for Image and Video Databases VI*, volume SPIE 3312, pages 150–161, 1997.

[92] B.C. O'Connor. Selecting key frames of moving image documents: A digital environment for analysis and navigation. *Microcomputers for Information Management*, 8(2):119–133, 1991.

[93] G. Pass and R. Zabih. Histogram refineement for content-based image retrieval. In *IEEE Workshop on applications of Computer Vision*, pages 96–102, 1996.

[94] A. Pentland, R.W. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254, 1996.

[95] T. Poggio and S. Edelman. A network that learns to recognize three-dimensional objects. *Nature*, 343(6255):263–266, January 1990.

[96] T. Pun and D. Squire. Statistical structuring of pictorial databases for content-based image retrieval systems. *Pattern Recognition Letters*, 17:1299–1310, 1996.

[97] G. Celeux E. Diday G. Govaert Y. Lechevallier H. Ralambondrainy. *Classification automatique des données*. DUNOD, 1989.

[98] R. Redner and H. Walker. Mixture densities, maximun likelihood and the em algorithm. *SIAM Review*, 26:195–239, 1984.

[99] A. Ribert. *Structuration évolutive de données : application à la construction de classifieurs distribués*. PhD thesis, Rouen, France, october 1998.

[100] A. Ribert, A. Ennaji, and Y. Lecourtier. A multi-scale clustering algorithm. *Vision interface*, pages 592–597, may 1999.

[101] R. Rickman and J. Stonham. Content-base image retrieval using color tuple histograms. *SPIE*, pages 2–7, 1996.

[102] B.D. Ripley. *Pattern Recongition and Neural Networks*. Cambridge University Press, 1996.

[103] C. Robert. L'analyse statistique bayésienne. *Economica*, 1992.

[104] R. Rosales. Recognition of human action using moment-based features. Technical Report Report BU 98-020, Boston University Computer Science, Boston, MA 02215, November 1998.

[105] Y. Rui, T. S. Huang, and S. Mehrotra. Constructing table-of-content for videos. *ACM Multimedia Systems Journal, Special issue Multimedia Systems on video librairies*, 1999. To appear.

[106] Y. Rui, T. S. Huang, and S. Mehrotra nad M. Ortega. Automatic matching tool selection using relevance feedback in mars. In *Second International Conference on Visual Information Systems*, pages 109–116, 1997.

[107] G. Saporta. *Probabilités, Analyse des Données et Statistique*. Eds Technip, Paris, 1990.

[108] B. Schiele. *Reconnaissance d'objets utilisant des histogrammes multidimensionnels de champs réceptifs*. Thèse de doctorat, GRAVIR – IMAG – INRIA Rhône–Alpes, July 1997.

[109] C. Schmid and R. Mohr. Mise en correspondance par invariants locaux. In *Actes du 10ème Congrès AFCET de Reconnaissance des Formes et Intelligence Artificielle, Rennes, France*, pages 236–245, 1996.

[110] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, May 1997.

[111] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, March 1978.

[112] T. Sikora. The mpeg-4 video standard verification model. *IEEE Trans. Circuits Systems Video Technol.*, 7(1):19–31, 1997.

[113] 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.

[114] C. Sminchisescu and B. Triggs. A robust multiple hypothesis approach to monocular human motion tracking. In *International Conference on Computer Vision (iccv'2001)*, Vancouver, Canada, 2001. Submitted by 4 December.

[115] M. Smith and T. Kanade. Video skimming for quick browsing based on audio and image characterization. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Puerto Rico, USA*, 1997.

[116] H. Stokman and T. Gevers. Photometric invariant region detection in multispectral images. In *Proceedings of the 12th Conference on Vision Interface, Trois Rivières. Québec*, pages 90–96, 1999.

[117] M. Stricker and A. Dimai. Color indexing with weak spatial constraints. In SPIE, pages 29–40, 1996.

[118] H. Sundaram and S. Chang. Determining computable scenes in films and their structures using audio-visual memory models. In *8th ACM International Conference on Multimedia*, pages 95–104, Los Angeles, CA, USA, November 2000.

[119] K.K. Sung and T. Poggio. Example-based learning for view-based human face detection. IEEE *Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.

[120] M.J. Swain and D.H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[121] M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Maui, Hawaii, USA*, pages 586–591, 1991.

[122] H. Ueda and al. Automatic structure visualization for video editing. In *Proc. InterCHI 93, ACM press, New York*, pages 137–141, 1993.

[123] P. Valtchev. Building classes in object-based languages by automatic clustering. *IDA'99*, pages 303–314, 1999.

[124] J. H. Ward. Hierarchical grouping to optimize an objectif function. *Journal of the American Statisticall Association*, 58:236–244, 1963.

[125] C. Wu. On the convergence properties of the em algorithm. *Annals of Statistics*, 1983.

[126] S. Yakowitz and J. Spragins. On the identifiability of finite mixtures. *Annals of Mathematics and Statistics*, 39:209–214, 1968.

[127] C.Y. Yang and J.C. Lin. Rwm-cut for color image quantization. *Comput and Graphics*, 20(4):577, 1996.

[128] B.L. Yeo and B. Liu. Rapid scene analysis on compressed video. *IEEE Trans. on Circuits and Systems for Video Technology*, 6(5):533–544, 1995.

[129] B.L. Yeo and M. Yeung. Retrieving and visualizing video. *Communication of the ACM*, 40(12):43–52, 1997.

[130] M. Yeung and B. Yeo. Segmentation of video by clustering and graph analysis. *Computer Vision and Image Understanding*, 71(1):94–109, July 1998.

[131] R. Zabih, J. Miller, and K. Mai. Feature-based algorithm for detecting and classifying scene breaks. In *ACM Mulimedia Conference*, pages 130–136, San Francisco, CA., Novembre 1995.

[132] H. Zhang, Y. Gong, and S. Smoliar. Automatic parsing of news video. In *Proc. of IEEE conference on Multimedia Computing and Systems*, Boston, USA, 1994.

[133] H. J. Zhang, C. Y. Low, S.W. Smoliar, and J.H. Wu. Video parsing, retrieval and browsing: An integrated and content-based solution. *ACM Multimedia*, pages 15–24, 1995.

[134] Y. Zhuang, Y. Rui, T. S. Huang, and S. Mehrotra. Adaptive key frame extraction using unsupervised clustering. In *Proc. of IEEE conf. on Image Processing*, pages 866–870, Chicago, IL, October 1998.

Construction et Présentation des Vidéos Interactives

L'arrivée de la norme MPEG-7 pour les vidéos exige la création de structures de haut niveau représentant leurs contenus. Le travail de cette thèse aborde l'automatisation de la fabrication d'une partie de ces structures. Comme point de départ, nous utilisons des outils de segmentation des objets en mouvements. Nos objectifs sont alors : retrouver des objets similaires dans la vidéo, utiliser les similarités entre plans caméras pour construire des regroupements de plans en scènes. Une fois ces structures construites, il est facile de fournir aux utilisateurs finaux des outils de visualisation de la vidéo permettant des navigations interactives : par exemple sauter au prochain plan ou scène contenant un personnage.

La difficulté principale réside dans la grande variabilité des objets observés : changements de points de vues, d'échelles, occultations, etc. La contribution principale de cette thèse est la modélisation de la variabilité des observations par un mélange de densités. La théorie du mélange gaussien est employée dans cette approche. Cette modélisation permet de capturer les différentes apparences intra-plan de l'objet suivi et de réduire considérablement le nombre des descripteurs de bas niveaux à indexer par objet suivi.

Autour de cette contribution se greffent des propositions qui peuvent être vues comme des mises en oeuvre de cette première pour différentes applications : mise en correspondance des objets suivis représentés par des mélanges gaussiens, fabrication initiale des catégories de tous les objets présents dans une vidéo par une technique de classification non supervisée, extraction de vues caractéristiques et utilisation de la détection d'objets similaires pour regrouper des plans en scènes.

Mots clefs : Vidéo hyperliée, MPEG-7, Reconnaissance et classification d'objets, Modélisation de la variabilité, Modèles de mélange gaussien, Navigation interactive, Structure de la vidéo.

Constructing and Browsing of Interactive Videos

The arrival of the MPEG-7 standard for videos requires the creation of high level structures representing their content. The work of this thesis approaches the automatic building of a part of these structures. As a starting point, we use the tools for segmentation of moving objects. Our objectives are then to find similar objects in the video and subsequently use the similarities between camera shots to group shots into video scenes. Once these structures have been built, it is easy to provide video visualization tools for the end users which permit interactive navigation like jumping to the next shot or scene containing a person.

The main difficulty lies in the great variability of observed objects: changes in point of view, scales, collusions, etc. The principal contribution of this thesis is the modeling of the variability of observations by a mixture of densities based on the Gaussian mixture theory. This modeling captures various intra-shot appearances of a tracked object and considerably reduces the number of low-level descriptors to be indexed by each tracked object.

The proposed formulation led to an implementation designed for different applications: matching of tracked object models represented by Gaussian mixtures, initial building of categories of all objects present in a video by a non-supervised classification technique, extraction of characteristic views and use of detected similar objects for grouping shots into scenes.

Keywords : Hyperlinked video, MPEG-7, Object recognition and classification, Variability modeling, Gaussian mixture models, Interactive video navigation, Video structure.

Thèse préparé dans le laboratoire GRAVIR–IMAG et INRIA Rhône-Alpes, au sein de l'équipe MOVI.
INRIA Rhône-Alpes, 655, Av. de l'Europe, ZIRST, 38330 Montbonnot Saint-Martin.

Annexe A

Propriétés de l'algorithme EM

Saluer ce qui va pour mieux dire
ce qui ne va pas.

Cet annexe présente une étude sur la croissance de la vraisemblance et la convergence de l'algorithme itératif EM et ses variantes. On conserve ici les mêmes notations utilisées dans la section 3.6.3.2.

A.1 Croissance de la vraisemblance

Cette procédure a pour propriété fondamentale de faire croître la vraisemblance des paramètres au cours des itérations [60]. Pour le voir, notons d'abord que

$$\begin{aligned} Q(\theta,\theta^m) &= \Big\langle \log f(z \mid y,\theta) + \log f(y \mid \theta) \mid y,\theta^m \Big\rangle \\ &= \underbrace{\log f(y \mid \theta)}_{L(\theta)} + \underbrace{\Big\langle \log f(z \mid y,\theta) \mid y,\theta^m \Big\rangle}_{H(\theta,\theta^m)} \end{aligned}$$

$$\Longrightarrow L(\theta) = Q(\theta,\theta^m) - H(\theta,\theta^m)$$

Or, la fonction $H(\theta,\theta^m)$ possède une propriété intéressante :

$$\forall\theta \quad H(\theta,\theta^m) \leq H(\theta^m,\theta^m) \tag{A.1}$$

qui découle de l'inégalité de Jensen [31]. On peut également retrouver la relation A.1 en remarquant que

$$H(\theta,\theta^m) - H(\theta^m,\theta^m) = \sum_{z\in\mathcal{Z}} f(z \mid y,\theta^m) - \log \frac{f(z \mid y,\theta)}{f(z \mid y,\theta^m)}$$

où $\mho$ désigne l'ensemble des valeurs que peut prendre la vecteur aléatoire z (cas non continu). Comme $\forall x \in I\!R, \log(x) \leq x - 1$, on peut dire que $\forall z \in \mho$,

$$\log \frac{f(z \mid y, \theta}{f(z \mid y, \theta^m} \leq \frac{f(z \mid y, \theta}{f(z \mid y, \theta^m} - 1$$

donc

$$H(\theta, \theta^m) - H(\theta^m, \theta^m) \leq \underbrace{\sum_z f(z \mid y, \theta)}_{1} - \underbrace{\sum_z f(z \mid y, \theta^m)}_{1} = 0$$

d'où l'inégalité A.1. De plus, comme $\log(x) = x - 1$ si et seulement si $x = 1$, il y a égalité si et seulement si $\forall z \in \mho, f(z \mid y, \theta) = f(z \mid y, \theta^m)$. L'inégalité A.1 permet d'écrire que

$$H(\theta^{m+1}, \theta^m) \leq H(\theta^m, \theta^m).$$

Comme par ailleurs θ^{m+1} maximise $Q(\theta, \theta^m)$, on a

$$Q(\theta^{m+1}, \theta^m) \geq Q(\theta^m, \theta^m)$$

d'où

$$L(\theta^{m+1}) = Q(\theta^{m+1}, \theta^m) - H(\theta^{m+1}, \theta^m) \geq Q(\theta^m, \theta^m) - H(\theta^m, \theta^m) = L(\theta^m)$$

Notons que cette propriété de croissance de la vraisemblance reste vérifiée si l'on se contente de mettre à jour les paramètres θ^{m+1} pour que $Q(\theta^{m+1}, \theta^m) \geq Q(\theta^m, \theta^m)$.

A.2 Convergence de EM

Le concept de EM se base sur le fait que la valeur θ^* maximisant la vraisemblance $L(\theta)$ constitue un point fixe de la fonction F qui met à jour les paramètres d'une itération de EM. Cette fonction peut être définie comme

$$F : \theta^0 \mapsto \arg\max_\theta Q(\theta, \theta^0).$$

En effet, en posant

$$\theta^* = \arg\max_\theta L(\theta)$$

on remarque que d'après l'équation A.1,

$$\theta^* = \arg\max_\theta H(\theta, \theta^*)$$

donc

$$\begin{aligned}\theta^* &= \arg\max_\theta \underbrace{L(\theta) + H(\theta, \theta^*)}_{Q(\theta,\theta^*)} \\ &= F(\theta^*).\end{aligned}$$

Dempster et al. [31] montre que les points fixes de l'algorithme EM correspondent à des points stationnaires de la vraisemblance. Quant à la convergence de l'algorithme, elle a été étudiée pour des classes de problèmes particuliers. Une étude détaillée de la convergence de EM est mené par Wu [125].

Annexe B

Séquence "Dances with Wolves"

Saluer ce qui va pour mieux dire
ce qui ne va pas.

Les plans vidéo illustrés dans les figures B.1 et B.2 sont extraits du film "Dances with wolves" par la méthode de Ardebilian et al. [4]. Cette séquence a été préparée dans le laboratoire Heudiasyc de l'Université de Technologie de Compiègne. Ne possédant que des images médianes des 70 plans, certaines images contenant des objets d'intérêts ont été segmentées manuellement.

Cette séquence a été utilisée dans la validation expérimentale de l'approche de macro-segmentation proposée dans le chapitre 7. Une macro-segmentation manuelle de cette séquence permet d'identifier les sept scènes suivantes :

- scène 1 du plan 1 jusqu'à le plan 29
- scène 2 comporte seulement le plan 30
- scène 3 du plan 31 jusqu'à le plan 47
- scène 4 du plan 48 jusqu'à le plan 51
- scène 5 comporte seulement le plan 52
- scène 6 du plan 53 jusqu'à le plan 67
- scène 7 du plan 68 jusqu'à le plan 70

FIG. B.1: *Plans de la séquence "Dances with wolves"*

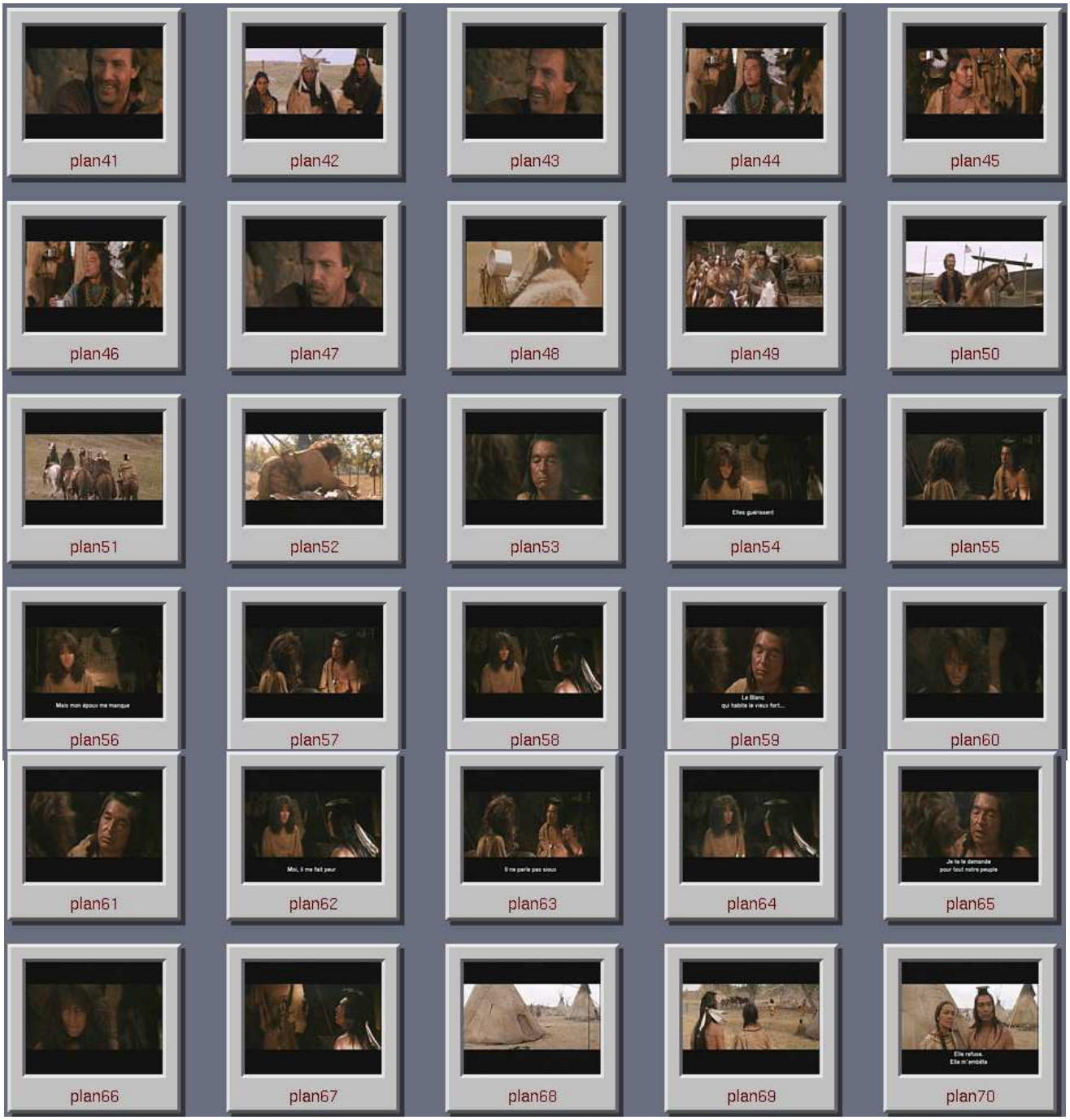


FIG. B.2: *Plans de la séquence "Dances with wolves" (suite)*