Laboratoire d'InfoRmatique en Images et Systèmes d'information

Institut National de l'Audiovisuel

THÈSE

Compréhension de systèmes d'extraction d'objets dans la vidéo sous l'angle de l'adaptation

Présentée et soutenue publiquement le :

23 Mars 2006

Par

Rémi Landais

Pour l'obtention du :

Doctorat de l'Institut National des Sciences Appliquées de Lyon
(spécialité informatique)

Décerné par :

Ecole Doctorale en Informatique et Information pour la Société

Composition du jury

Président :	J-P. Haton, LORIA Nancy
Rapporteurs :	C. Garbay, TIMC-IMAG Grenoble M. Revenu, Université de Caen
Directeurs :	J-M. Jolion, INSA de Lyon L. Vinet, Institut National de l'Audiovisuel

Mis en page avec la classe thloria.

Remerciements

Cette thèse s'est déroulée dans le cadre d'une bourse CIFRE entre l'Institut National de l'Audiovisuel (INA), au sein de l'équipe Description des Contenus, et le laboratoire du LIRIS. Laurent Vinet, à l'INA, et Jean-Michel Jolion au LIRIS se sont ainsi partagé la tâche de l'encadrement de ce doctorat.
Ils ont tous deux contribué grandement à l'aboutissement de ces travaux. Par leur disponibilité, la qualité de leur écoute, leur expérience, ils ont su, en me laissant toute latitude dans mes recherches, orienter ces dernières de la meilleure des façons. Pour autant, il serait réducteur de limiter les bienfaits de cette collaboration au seul aboutissement de la thèse. Les échanges au cours de ces trois années de thèse sortirent ainsi souvent du seul cadre professionnel et toutes ces autres discussions me furent tout aussi profitables. C'est donc en toute amitié que je remercie Laurent et Jean-Michel pour l'accompagnement qui a été le leur pendant cette thèse.
Bien entendu, les laboratoires ne se limitent pas aux seuls directeurs de thèse. Je salue donc ici mes compagnons de voyage de l'INA (dans le désordre) : Thomas, Raphaël, Antoine, Jean-Calude, Véronique, Vincent, Jean, Fabrice, Odile, Jérôme, Jean-Philippe, Ludovic, Nicolas, Hugo, Laurent... Ceux de Lyon : Christian, Rémi, Mickael ... De Rennes : Julien, Muriel... Avec une mention spéciale pour Alexandre, seul travailleur des lointaines contrées de l'Essonne et qui n'hésitait pas à braver la ligne A pour rejoindre Bry-Sur-Marne les Vendredis pluvieux...

Jusqu'à présent, je pense que 6 personnes ont lu mon manuscrit. Parmi ces personnes, j'ai déjà cité mes deux directeurs, Laurent et Jean-Michel qui l'ont lu et relu, corrigé, raturé, annoté... Viennent ensuite mes deux rapporteurs et mon président de Jury : Mme Catherine Garbay, Mme Marinette Revenu et M Jean-Paul Haton. Je les remercie tous les trois d'avoir accepté de faire parti de mon jury, d'avoir relu avec attention ce manuscrit et surtout de m'avoir permis, grâce à l'ensemble de leurs remarques, de parvenir, je pense, à améliorer ce dernier. Par l'attitude bienveillante qu'ils ont témoignée à l'égard de mes travaux, autant dans leurs compliments que dans leurs critiques, les membres du jury ont à mon sens rempli bien plus que la seule tâche d'évaluation qui leur était confiée.

Il serait malhonnête de faire croire que ma thèse m'a tenu reclus pendant trois ans. Pour tous les nombreux moments agréables passés ailleurs que derrière un PC (c'est à dire tout ceux passés, bien entendu, à lire du Russe, à l'opéra ou à une exposition d'art contemporain), je tiens donc à remercier : Marie, Martin, Kévin, Aurélie, Henri, Aurélien, Benjamin, Juliette, Charby (il en fallait bien un avec un surnom), Claire, Marlotte, Flox, Meriem, Cléo, Slim, Grégoire, Raphaël, Nolwenn, Aaron, Julie, Lorcan, Patricia, Louise...
Last but not least, je tiens à remercier profondément ma famille qui m'a énormément apporté (et ce n'est rien de le dire) durant ces 26 dernières années : ma mère, mon père, ma soeur, mes grands-parents. Merci pour tout... Tout comme ma "belle" famille de Créteil, de Marseille ou encore de Toulouse.

Seuls 5 des 6 lecteurs providentiels de mon manuscrit ont été dévoilés... Le dernier lecteur mystère de ce manuscrit n'est autre que mon indispensable compagne, Chloé, qui a su me porter jusqu'à la soutenance en me rappelant, dès que le besoin s'en faisait sentir, la célèbre maxime : "Un bon manuscrit est un manuscrit envoyé aux rapporteurs". Pour tout cela et bien plus encore : merci beaucoup Chloé.

Table des matières

Table des figures

Première partie

Positionnement

1

Documentation audiovisuelle et extraction d'objets : de la nécessité de mettre en place une méthodologie d'adaptation

Sommaire

"*Aux temps de l'internet, du DVD et du tout-numérique, le volume des documents numériques disponibles ne cesse d'augmenter, nécessitant de mettre en place des méthodologies intelligentes de compréhension et de gestion des fonds ...* " Ce sage constat occupe souvent les premières lignes des articles relatifs au domaine de l'image ou de la vidéo. La communauté scientifique s'est ainsi trouvé un cheval de bataille fédérateur, source à de multiples recherches dont il semble qu'elle ne soit pas amenée à être tarie avant longtemps tant les difficultés rencontrées s'avèrent importantes.

Une part de ces difficultés naît de la complexité intrinsèque des systèmes de vision et de la tâche finale qui leur est désormais attachée, à savoir d'aider à la gestion des fonds documentaires (images ou vidéos). L'enjeu de cette introduction est alors de détailler ces obstacles, sous l'angle applicatif de la documentation audiovisuelle d'une part et d'autre part, sous l'angle scientifique des systèmes d'extraction d'objets dont l'objectif est justement d'aider à la documentation en

dégageant du flux certains objets particuliers. Nous verrons alors de quelles insuffisances souffrent ces systèmes, notamment au niveau de la phase de modélisation des objets et serons finalement amenés à poser le postulat d'une nécessaire **adaptation** dont une nouvelle méthodologie sera présentée dans la suite de ce manuscrit. Une brève présentation de l'Institut National de l'Audiovisuel, au sein duquel s'est effectuée cette étude servira par ailleurs d'introduction à notre propos.

1.1 Contexte Industriel

La présentation que nous proposons ici de l'Institut National de l'Audiovisuel n'est pas exhaustive. Pour de plus amples informations, il est conseillé au lecteur de visiter le site internet de l'INA (`www.ina.fr`). Pour simplifier, on peut distinguer à l'INA deux entités, lesquelles sont respectivement en charge des deux principales missions de l'INA :

- *La direction des Archives* conserve et exploite les documents audiovisuels diffusés sur les chaînes publiques hertziennes de radio et de télévision.
- *L'Inathèque de France* collecte et conserve, relativement à la législation, les documents sonores et audiovisuels radiodiffusés ou télédiffusés.

Conformément à l'évolution des formats des documents audiovisuels, les chiffres, révélateurs de l'ampleur de la tâche de conservation dont est en charge l'INA, sont eux-mêmes de deux types, les uns référant aux volumes de documents au format numérique, les autres référant aux volumes des documents au format analogique. En se limitant aux documents télédiffusés nous obtenons le tableau 1.1.

	Archives professionnelles	**Inathèque de France**
Nombres d'heures	535 000	430 000
Pourcentage de documents numérisés	21%	51%
Nombres d'heures collectées par an	37 922	113 376

TAB. 1.1 – Volumes gérés par les "institutions" de l'INA

De tels volumes soulèvent inévitablement de nombreux problèmes. Au delà de la simple "veille technologique" relative aux problèmes de captation du flux et du stockage (quels supports de stockage ?, etc), ce sont de nouvelles thématiques de recherche qui apparaissent, thématiques en relation avec des champs disciplinaires extrêmement différents : de la sociologie au traitement du signal en passant par la représentation des connaissances et l'informatique. L'INA se veut alors le reflet de ses évolutions, la Direction de la Recherche donnant ainsi l'image de cette diversité dans la nature des thèmes qui y sont abordés.
Pour autant, nous retiendrons principalement de cette présentation que les volumes disponibles à l'INA font de cette dernière une plateforme d'expérimentation unique en son genre, mettant la plupart du temps à mal des systèmes dont la conception repose trop souvent sur des corpus de test extrêment petits au regard de ces volumes. Par ailleurs, un dernier point important relève de l'organisation des documents autour de collections homogènes. Le mode d'intégration de ces contraintes volumiques et organisationnelles dans la méthodologie proposée sera détaillé par la suite.

1.2 De la manipulation des images à la documentation

Le traitement de l'image n'a pas toujours été au service d'instituts tels que l'INA. Avant de s'atteler à la tâche de l'adaptation de systèmes d'extraction d'objets, il paraît ainsi naturel de revenir aux sources d'un domaine vieux maintenant de plus de cinquante ans, pour en rappeler les évolutions et resituer le contexte scientifique de notre étude.

Le traitement de l'image est la science ayant pour objet la manipulation des images numériques. Si l'objet d'étude (l'image) est clairement énoncé dans cette définition, les délimitations théoriques autant que pratiques de ce domaine demeurent floues. Il est courant d'avoir recours à la distinction entre le *traitement* et l'*analyse* d'images dans le but de circonscrire le champ d'étude trop général de la *manipulation d'images*. A ses terminologies nous pourrons parfois préférer celles de *prétraitement* et de *compréhension*, exprimant à notre sens plus clairement les objectifs de chacun de ces deux domaines.
La différence essentielle entre le prétraitement et la compréhension réside dans la négation de la composante sémantique de l'image par le prétraitement. Dans ce domaine, reposant essentiellement sur des méthodes d'analyse numérique et de traitement du signal, les images sont des matrices de pixels ou des signaux 2D, c'est à dire des objets essentiellement *numériques* ou *analytiques* sur lequel des opérateurs sont appliqués. Dans le cas de la compréhension, l'intérêt réside essentiellement dans le sens des images, dans les informations qu'elles véhiculent. Dès lors, les champs applicatifs respectifs de ces deux domaines sont profondément différents bien que l'objet d'étude reste le même.
Dans le cas du prétraitement, les images sont transformées en d'autres images. Des opérateurs sont crées pour compresser, améliorer ou restaurer. Certaines informations peuvent être mises en valeur (augmentation de contraste, seuillage, ...), mais l'intégralité de l'information sémantique extractible de l'image demeure *dissimulée* dans la valeur de ses pixels, le célèbre *fossé sémantique* sur lequel nous reviendrons plus loin, demeure infranchi. Le prétraitement se révèle finalement être au service des applications de plus haut niveau relatives à la compréhension. C'est ainsi que dès les années soixante, les premiers opérateurs de compression ou d'amélioration sont implémentés pour permettre aux médecins d'établir plus facilement un diagnostic ou pour permettre aux scientifiques de la NASA d'analyser plus facilement les clichés pris par les satellites [Jol87].
Le domaine de la compréhension des images a pour objectif de proposer une représentation intelligible de l'image dans un objectif donné. Cette représentation peut être une nouvelle image dans laquelle sont isolés des objets particuliers (par exemple des cellules déficientes). Elle peut aussi consister en un ensemble de mots clés, en une reconstruction en 3D de la scène, etc. La création de cette représentation passe par l'extraction d'informations de haut-niveau sur le contenu de l'image et/ou sa forme. A la différence du prétraitement dont les fondements reposaient essentiellement sur le traitement du signal, la compréhension des images fait intervenir un spectre plus large de connaissances scientifiques : le traitement du signal de nouveau, la représentation des connaissances, l'interaction homme-machine et surtout l'intelligence artificielle. Historiquement, c'est la vision artificielle qui fut l'un des premiers défis relevés durant les années 80... Malheureusement, les résultats obtenus aujourd'hui n'atteignent pas encore les espérances portées dans ce domaine de recherche il y a bientôt trente ans. L'écart entre le monde tel que nous le voyons et le monde tel que nous le modélisons est encore trop important. Le rêve de l'ordinateur doué de vision a certes fait long feu mais la compréhension d'images a néanmoins proposé dans des cadres applicatifs beaucoup plus restreints (par exemple en vision industrielle) des exemple de réussite indéniables. Une nouvelle fois, il apparaît que ce sont les applications qui sont au centre de la conception de nouveaux systèmes, au coeur de la réflexion théorique du domaine. Ce sont en effet

les applications qui induisent quelles informations doivent être extraites des images. La tâche du concepteur d'un système consiste alors essentiellement à reformuler la problématique (contenue dans le cadre applicatif) pour définir une modélisation des informations à extraire suffisamment proche du signal pour que des opérateurs de prétraitement adéquats puissent être choisis.
Malgré ces différences, les domaines du prétraitement et de la compréhension possèdent un point commun indiscutable : leur dépendance commune aux avancées technologiques dans le domaine du hardware. Traiter ou comprendre une image sur un ordinateur était il y a 30 ans l'apanage des laboratoires d'importance et à plus forte raison, des industriels pour des raisons évidentes de capacité à acquérir des machines suffisamment puissantes. L'explosion des capacités des ordinateurs a accéleré les traitements et aboutit à une formidable démocratisation de l'image numérique.
Pour autant, ce sont dans le même temps de nouvelles problématiques qui se sont dessinées : la gestion des images (ou des vidéos) disponibles dans des volumes désormais colossaux posent de nombreux problèmes tant au niveau de la définition d'opérateurs de prétraitement adaptés que dans la conception de systèmes de compréhension d'images. Les questions de l'organisation des bases, de leur potentielle compréhension, de leur documentation deviennent particulièrement complexes du fait des volumes de données qu'il convient de traiter : la nature des documents varie énormément et impose ainsi une réflexion beaucoup plus approfondie sur la spécialisation des systèmes autrefois dédiés à des classes d'images similaires (images satellites, images médicales,...).
Parmi les tâches les plus ambitieuses envisagées se remarque celle de la documentation audiovisuelle. Celle-ci implique de trouver une représentation de très haut niveau des documents. La difficulté est d'autant plus grande que la nature des informations qu'il convient d'extraire est dépendante non seulement de l'application finale visée mais aussi de la nature propre des documents considérés.

1.3 La documentation audiovisuelle

L'aide à la documentation audiovisuelle est l'un des nouveaux défis pour le *prétraitement* et de façon encore plus marquante pour la *compréhension* des images. Nous proposons dans cette partie de dresser un panorama de la documentation audiovisuelle nous permettant de mettre à jour les enjeux de cette application en insistant particulièrement sur les points durs de la mise en place d'une automatisation des procédés la constituant.

1.3.1 Les spécificités du document audiovisuel

Le document audiovisuel (document AV) est *temporel* et constitué d'*images et de sons*[1]. Ces spécificités sont majeures puisque le document audiovisuel ne peut pas être appréhendé selon la même méthodologie que les documents textuels. En effet, une structure immédiate de ces derniers est celle imposée par la typographie, par exemple les espaces entre les mots ou la ponctuation. Bien que l'unité physique consacrée du document audiovisuel soit l'image, il est imprudent, voire incorrect de considérer un document AV comme une simple séquence d'images : son sens, sa signification ne provient pas de l'aggrégation de sens unitaires assignés aux images mais bien d'un sens global issu d'une compréhension de la logique de construction des séquences d'images. Ces séquences d'images peuvent alors être des images isolées, des plans, ou même des scènes qui sont autant d'équivalents aux mots, lignes, paragraphes et chapitres des documents

[1] Il est à remarquer dès à présent que nous ne prenons pas en compte la bande sonore bien que sa valeur en tant que vecteur d'informations ait été consacrée lors des campagnes d'évaluation TRECVIDEO (`http://www-nlpir.nist.gov/projects/trecvid/`)

textuels.
Par la suite, la question se pose donc de définir un sens à un segment temporel issu d'un document audiovisuel. A la différence du document textuel, "l'image ne peut dire par elle-même ce qu'elle signifie et doit reposer sur une paraphrase langagière pour gagner l'intelligibilité qu'il lui manque" [Bac99]. Une représentation textuelle transverse des documents audiovisuels est donc nécessaire pour faciliter leur manipulation. Pour autant établir cette représentation est difficile : la façon dont les images sont perçues et analysées dépend d'une part de la subjectivité du documentaliste mais aussi du contexte applicatif dans lequel s'effectue la description. Nous présentons dans la partie suivante les applications les plus importantes de la documentation. Sans les décrire précisément pour l'instant, prenons pour cadre certaines d'entre elles pour illustrer notre propos et admettons de surcroît que l'objectif est de documenter l'image de la figure 1.1. Il existe différents **contextes de description** de cette image qui induisent des différences dans

FIG. 1.1 – Une photographie qu'il convient de décrire : quelles informations faut-il retenir ?

les informations retenues pour paraphraser la photographie. Voici quelques exemples de ce que peuvent être de tels contextes :

- Constituer un corpus illustrant les techniques utilisées en photographie : nature de l'angle, de la profondeur de champ, des choix concernant l'illumination de la scène, (**thématisation**)
- Constituer un corpus des oeuvres de Cartier-Bresson : qu'est-ce qui permet d'identifier que la prise de vue a été effectuée par Cartier-Bresson ?, Comment resituer cette prise de vue dans le contexte historique de l'oeuvre du photographe ? (**thématisation**),
- Effectuer une description purement formelle du contenu de l'image (**indexation**)
- Est-ce que cette photographie doit être retenue pour une présentation succinte de l'oeuvre de Cartier-Bresson ? (**navigation**)

La description est donc une tâche complexe, profondément liée à l'application finale visée et à la subjectivité de l'annotateur. L'aide à la documentation se trouve ainsi d'autant moins facilitée. Comme nous l'avons déjà mentionné, la première phase de conception d'un système de compréhension consiste à reformuler le cadre applicatif en termes d'informations qu'il est nécessaire d'extraire. La notion de connaissances *a priori* est alors particulièrement prégnante puisqu'une contextualisation précise permet de qualifier beaucoup plus précisément la nature de ces informations. La question de l'inclusion de telles connaissances dans le système se trouve alors soulevée. Dans la partie suivante, nous présentons certaines des applications les plus courantes liées à la documentation. Pour chacune de ces applications, la question de leur automatisation sera abordée.

1.3.2 Quels besoins pour la documentation audiovisuelle ?

La documentation audiovisuelle englobe un ensemble de procédés plus large que la seule tâche de description. Si on se réfère au cycle de vie d'un document audiovisuel (*Conception, Production, Exploitation*), il apparaît que la documentation peut être produite et mobilisée à chacun des stades de son cycle de vie.
La documentation regroupe alors non seulement toutes les tâches relatives à la production des descriptions mais encore toutes les tâches touchant à la mobilisation de ces mêmes descriptions. Nous proposons ci-dessous une liste non exhaustive des tâches de documentation. Pour chacune de ces tâches, l'automatisation est envisagée en présentant quelques systèmes existants et quelques réflexions plus prospectives.

1. **la segmentation** : le document est divisé en N segments temporels selon un critère d'homogénéité choisi en fonction de l'application finale. L'automatisation de cette tâche repose sur la définition d'une mesure de proximité entre certaines unités temporelles qui peuvent être le cas échéant fusionnées ou segmentées de nouveau. La mesure de similarité est plus ou moins difficile à définir. Dans le cas de la segmentation d'un journal télévisé en plateaux/reportages, des informations colorimétriques peuvent être suffisantes [Pol03]. Au contraire, dans le cadre d'une segmentation en scènes, les critères d'homogénéité utilisés deviennent plus complexes et sont généralement multimodaux [PLE01].

2. **la description** : essentiellement manuelle dans les instituts de conservation des archives audiovisuelles comme l'Institut National de l'Audiovisuel, la description consiste à remplir des formulaires (*notices*) contenant les informations primordiales sur le document. La figure 1.2 montre un extrait d'une telle notice.

   La notice est pré-documentée grâce aux informations fournies par les diffuseurs. Les champs relatifs au résumé et aux mots clefs sont laissés à la charge des documentalistes. Automatiser le remplissage de ces parties de la notice nécessite donc d'acquérir des informations de très haut niveau sur le document. Il est par exemple envisageable de segmenter dans un premier temps le document en *histoires* (c'est à dire en segments homogènes d'un point de vue sémantique) puis de qualifier chacun des segments obtenus. Pour autant, il est illusoire de penser que l'ordinateur puisse suppléer à la difficulté du travail des documentalistes. La production automatique d'un texte structuré tel que le résumé relève pour l'instant de l'utopie. Néanmoins, il est possible d'aider à sa rédaction en extrayant les informations suivantes :
   - localisation de la scène (intérieur/extérieur, ville/paysage, ...)[FMC99, Sav02, VFJZ01]
   - détection d'objets particuliers (visages, textes, ...)
   - transcription de la bande sonore [PMK+04]

   L'enjeu est alors de parvenir à fusionner les informations extraites des différentes modalités du document ([WCCS04]).

3. **la recherche** : l'aide à la recherche de documents est sans aucun doute la tâche sur laquelle la communauté scientifique s'est le plus penchée. La conception, l'organisation (informatique), le parcours des bases de données dédiées aux images et à la vidéo ainsi que la conception de langages de requêtes adaptés sont autant de thématiques de recherche contigües à celle de la définition de mesures de similarités entre documents. Dans le domaine de la recherche par le contenu, le concept d'indexation est particulièrement pregnant. Selon [Bac01], l'indexation se définit comme "le procédé d'analyse de documents dans le but de produire une représentation conceptuelle riche, exprimée dans un langage contrôlé". Il existe donc un écart entre les informations produites par une analyse automatique des documents

```
Numéro: 654681.001
Numéro DL: DL T 19970825 FR2 004.001
Sélection DL: 0
Titre collection: F2 le journal 20H00
Titre propre: F2 le journal 20H00 : [émission du 25 Août 1997]
Société de programmes: France 2
Chaîne de diffusion: France 2
Canal: Réseau 2
Producteurs: Producteur, Paris : France 2, 1997
Nature de production: Production propre
Statut de diffusion: Première diffusion
Extension géographique: National
Date de diffusion: 25.08.1997
Jour: lundi
Heure de diffusion: 19:59:17
Heure de fin de diffusion: 20:36:07
Durée: 00:36:50
Type de description: Emission composite
Genre: Journal télévisé
Médiamétrie: Information, journal national
Générique: REA,Leroux Jean Pierre;PRE,Bilalian Daniel

Sommaire:

1. [Rentrée scolaire : Haute Saône] à 20:00:21:00 - 00:01:53:00
Jean Bernard Schmidt. - France 2 RENTREE DES CLASSES EN HAUTE
SAONE. Aujourd'hui a eu lieu la rentrée scolaire pour 810 écoliers
de Haute Saône qui ont adopté un nouveau rythme scolaire : c'est
la semaine de 4 jours ou la semaine de 5 jours, avec 2 après-midi
libres. C'est une manière d'avoir du temps libre pour des
activités extra-scolaires, sportives ou culturelles.

2. [Prix du cartable] à 20:02:28:00 - 00:01:41:00 Damien Theveno.
- France 2 A une semaine de la rentrée scolaire, la famille
TREMIER s'est rendu au supermarché de Chartres pour effectuer les
premiers achats. La facture s'élève à 950 francs.

...

Doc. d'accompagnement: Conducteur
Couleur: Couleur
Titre matériel: [F2 du 25 août 1997 de 19h59 à 20h50]
Matériel: BETA SP : 1 élément, Parallèle antenne, Couleur, MONO,
          Définition : 625 lignes, Format : 1/2 pouce, Procédé : Béta,
          Signal : Analogique, Standard couleur : SECAM
```

FIG. 1.2 – Extrait d'une notice descriptive de l'INA relative à un journal télévisé

et l'indexation : une reformulation est nécessaire.
La recherche d'un document est initiée par une requête. Celle-ci est exprimée dans un langage formel (qui diffère selon la nature de la base : SQL, XQuery,...). Dans ce cas, le traitement des documents intervient en amont pour extraire des informations utilisées pour compléter les notices sur lesquelles s'effectue la recherche. Cette dernière peut aussi être initiée par une requête visuelle (une image) auquel cas on parle de recherche par le contenu [SWS+00]. De façon parallèle, les méthodes à base de signatures [JFB03] permettent de la même façon de rechercher si une image (ou un document audiovisuel) est contenu dans une

base. Dans ce dernier cas, les caractéristiques choisies pour décrire une image se doivent d'être les plus robustes possible aux modifications (orientation, occlusion, changement de luminance, ...).

4. **la navigation** : l'enjeu est de proposer un mode de parcours simplifié (donc généralement plus rapide) des documents (ou de la base de documents). D'une part le (ou les) documents doivent être représentés sous une forme qui permette d'appréhender leur contenu rapidement. D'autre part, des interfaces sont conçues pour naviguer facilement dans ces ensembles de représentations [PP00, Shi03, AS94]. La difficulté vient généralement de la taille des bases de documents (parfois plusieurs centaines de milliers de documents), de l'aspect "3D" des documents audiovisuels (deux dimensions spatiales et le temps), ou encore du choix des axes de navigation (qui peuvent être multiples : auteurs, dates, contenu, forme...). Si il existe de nombreux systèmes de navigation dans des bases d'images donnant satisfaction, trouver une solution équivalente pour les documents audiovisuels demeure un problème de recherche ouvert.
5. **la classification** : la tâche de *classification* consiste à regrouper les documents selon un critère de similarité qui peut être attaché au signal [CWK03] (par exemple, état de dégradation du document, la couleur, etc) ou plus généralement sémantique (on parle alors de *thématisation*).

Quelle que soit l'application considérée, le besoin d'une représentation intermédiaire des documents se fait sentir, celle-ci devant se situer entre le niveau des pixels, qui ne porte aucune sémantique, et celui de la description textuelle, difficile à manipuler. L'extraction d'objets saillants, d'un point de vue informatif, constitue alors une tâche d'importance relativement à la construction d'une telle représentation des images : savoir d'une image qu'elle contient des visages, du texte, des véhicules, etc, représente une indéniable progression vers le sens au regard de l'information portée par les seuls pixels. La partie suivante revient plus précisément sur le cas des systèmes visant à extraire de tels objets.

1.4 L'analyse d'image en aide à la documentation : l'intérêt de l'extraction d'objets

L'échec de la vision artificielle dans les années 80 est une illustration parfaite de la difficulté à franchir le fossé sémantique (ou sensoriel) [SWS+00], existant entre la représentation physique des images manipulées par les systèmes (en tant que matrices de pixels) et leur représentation sémantique, généralement sujette à subjectivité.
Comme nous l'avons déjà expliqué, une représentation textuelle des documents audiovisuels (en texte naturel comme le champ résumé des notices ou encore dans un langage structuré dédié tel que le MPEG7 ou plus récemment FDL [CCR05]) permet de faciliter l'ensemble des tâches de documentation (recherche, ...). L'aide à la production d'une telle représentation est une tâche extrêmement complexe du fait même de l'existence du fossé sémantique. La méthode la plus immédiate pour contourner le problème est de concevoir des systèmes extrêmement spécialisés dont l'objectif est d'extraire des informations facilement manipulables dans le cadre de la description. C'est le cas de l'extraction d'objets particuliers dans l'image. La nature des objets qu'il est intéressant d'extraire est conditionnée par l'application finale envisagée qui se confond généralement avec la nature des documents : un ballon de football ou une balle de tennis dans les documents sportifs [WP03] ; des véhicules, des situations anormales dans le cadre de l'aide à la conduite [SBM05] ou de la surveillance [KGCM04], etc. Les objets sont au coeur du processus de compré-

hension de l'image et *a fortiori* des vidéos. Ceci se manifeste autant dans l'intérêt porté par la communauté à cette problématique qu'à la volonté affichée par le groupe MPEG d'appréhender l'image par les objets qu'elle contient (norme MPEG-4[2], récemment retenue comme norme de compression de la TNT).
Parmi les objets les plus fréquemment abordés par la communauté scientifique, les visages et les textes suscitent un intérêt particulier. Le site *Google Scholar*[3], moteur de recherche Google dédié à la recherche scientifique permet d'obtenir les histogrammes de la figure 1.3 montrant l'augmentation continue du nombre de publications du domaine.

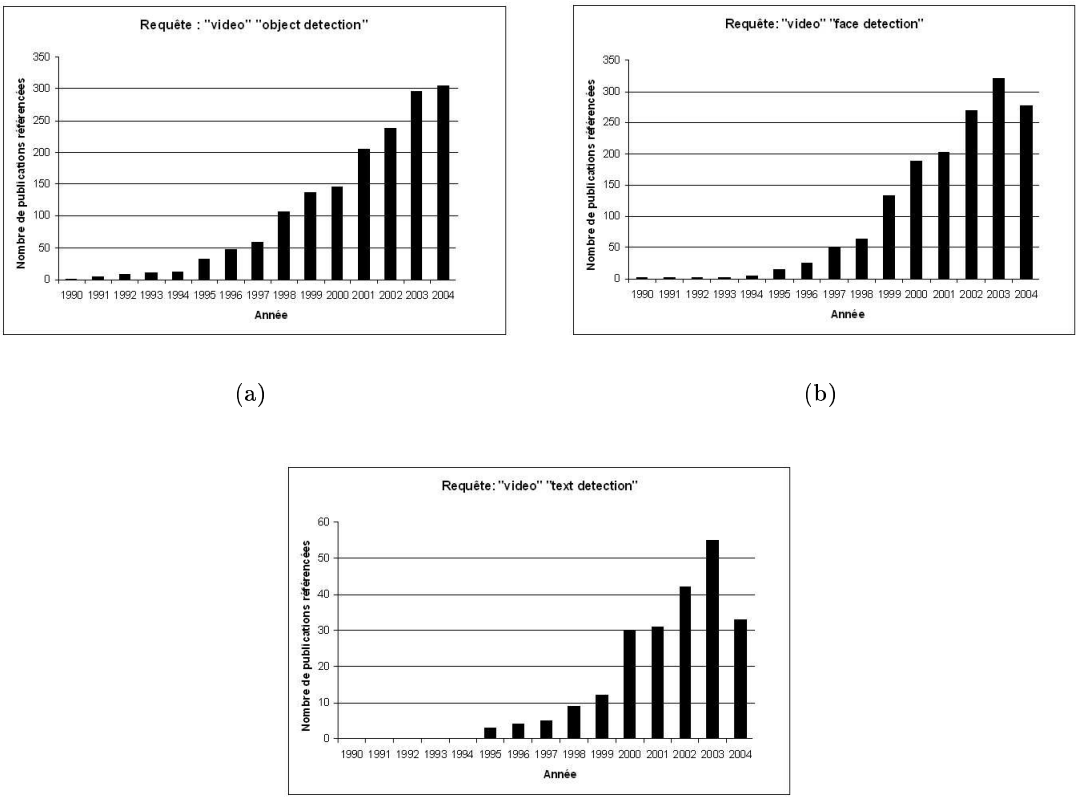


FIG. 1.3 – Répartition temporelle des publications sur la detection d'objets dans la vidéo

En vue d'aider à la documentation, les visages et les textes constituent des objets particulièrement intéressants dans la mesure où ce sont des objets que nous qualifierons de *traductibles*.

Définition 1. *Un objet vidéo est dit* ***traductible*** *si il existe des procédés permettant de produire une identification textuelle de celui-ci.*

Les procédés utilisés pour obtenir une telle identification textuelle relèvent de la reconnaissance des visages [ZCPR03] ou la reconnaissance de caractères [IOO91] (domaine communément considéré comme déclencheur de la recherche en reconnaissance de formes). L'existence de tels systèmes (même imparfaits) permet d'obtenir suite à la détection des objets (visages ou textes),

[2] `http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm`
[3] `http://scholar.google.com/`

des informations textuelles de très haut-niveau. Par ailleurs, le suivi de ces objets dans le flux audiovisuel permet d'obtenir des informations temporelles exploitables dans le cadre de la segmentation (telle personne apparaît à l'écran entre telle et telle date). En outre, certains systèmes proposent même d'associer les résultats de la reconnaissance du texte avec la détection des visages en vue d'identifier facilement les personnes parlant à l'écran [SNK97].
L'intérêt documentaire à extraire des objets est donc établi. La partie suivante s'attachera à démontrer en quoi cette tâche est particulièrement complexe et posera les premiers jalons de notre thèse selon laquelle l'obtention de systèmes d'extraction d'objets effectifs nécessite la mise en place d'une méthodologie d'*adaptation*.

1.5 De la conception à l'adaptation : un nécessaire cheminement

Il a déjà été reconnu par le passé dans le domaine du traitement de la parole que " [...] la grande variabilité du signal de parole [...] explique la difficulté de concevoir des algorithmes de reconnaissance robustes, capables de fonctionner pour un grand nombre de locuteurs. Si l'on se réfère à l'être humain, la solution se trouve à la fois dans la mise au point de procédures d'adaptation très évoluées et dans la recherche d'invariants, l'un complétant l'autre " [Hat85]. Ce point de vue est le nôtre et sera défendu tout au long de ce manuscrit. Hors du contexte du traitement de la parole, un retour est proposé dans cette partie sur les spécificités du domaine de l'extraction d'objets, nous autorisant à aboutir à la même préconisation : celle de l'adaptation.

1.5.1 Fonctionnement des systèmes d'extraction d'objets dans les vidéos

Les systèmes d'extraction d'objets dans les vidéos s'organisent autour de la séquence de quatre tâches : la détection des objets, leur suivi, l'amélioration et la reconnaissance. Le point de départ de cette séquence consiste donc à isoler dans les images du flux les zones contenant les objets recherchés. La figure 1.4 illustre alors le fonctionnement général de tout système d'extraction d'objets à ce stade.

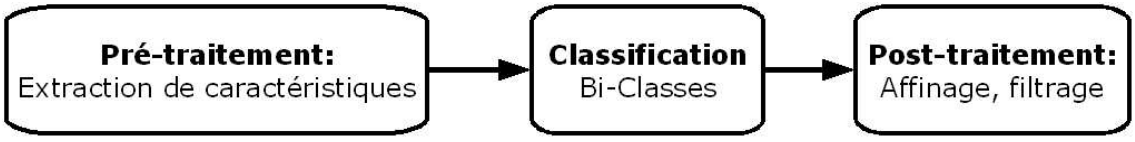


FIG. 1.4 – Représentation d'un système d'extraction d'objets sous la forme d'une séquence de trois tâches : pré-traitement, classification puis post-traitement

Dans un premier temps, une nouvelle représentation des données manipulées est proposée : les zones de l'image que l'on cherche à attribuer à la classe "objet" ou à la classe "autre" sont représentées selon un ensemble de caractéristiques. Les vecteurs produits représentant ces zones sont alors utilisés lors de la phase de classification, qui assigne effectivement chaque zone à l'une des deux classes considérées. Lors d'une dernière étape de post-traitement, les zones candidates extraites lors de la classification sont finalement affinées (leur délimitation spatiale est précisée) voire filtrées (suppression des zones assignées par erreur à la classe "objet", c'est à dire les fausses alarmes) selon un ensemble de caractéristiques potentiellement différent de celui utilisé lors de la phase de pré-traitement.

Les systèmes se différencient alors selon :

- **le mode de sélection des caractéristiques** : il peut soit reposer sur des connaissances *a priori* du concepteur, soit sur un mode de sélection (ou de construction par projection) automatique des caractéristiques à partir d'un ensemble plus important construit initialement. Pour cette seconde tâche on citera par exemple les méthodes SFS (Sequential Forward Selection), SBS (Sequential Backward Selection) et les algorithmes génétiques produisant parmi un ensemble de caractéristiques l'ensemble minimal permettant de produire les meilleurs résultats de classification ; ou les méthodes de construction de cet ensemble optimal de caractéristiques par projection comme la méthode de l'ACP [SBM04].
- **le mode de classification** : de la même façon, il convient de distinguer le mode manuel reposant généralement sur la détermination par une série d'essais de seuils délimitant les plages d'acceptation (pour la classe "objet") relativement aux différentes caractérististiques, du mode automatique reposant sur l'apprentissage (réseaux de neurones, méthodes bayésiennes, SVM, ... [DHS01]) et produisant à partir d'exemples les surfaces de décision permettant d'assigner une zone à une des deux classes considérées.

Pour autant, quel que soit le mode de conception des systèmes adopté, on remarquera que l'ensemble des systèmes produits induisent la construction d'un **modèle** des objets relativement aux caractéristiques choisies, modèle qui repose lui même sur la définition d'un **corpus de conception** (noté $CPS_{concept}$ par la suite) coïncidant exactement, dans le cas de l'apprentissage, avec le corpus d'appprentissage.

Etant donné la force du lien entre le système et le modèle qu'il induit, l'étude de ce modèle, et du corpus de conception lui étant associé, est nécessaire et servira de motivation majeure à la mise en place d'une méthologie d'adaptation.

1.5.2 Définition du modèle des objets

Un modèle est déterminé par les caractéristiques représentant les objets et par les plages d'acceptation relatives à ces différentes caractéristiques. Plus formellement, la définition suivante d'un modèle est adoptée :

Définition 2. *Le modèle d'un objet correspond à la projection de celui-ci dans un espace de caractéristiques déterminé, auxquelles sont assignées des plages de variation tôlérées.*

La formule 1.1 illustre alors la représentation d'un modèle $\mathcal{M}$ satisfaisant les contraintes de la définition précédente :

$$\mathcal{M} = \bigcap_{i=1}^{N} (f_i \in E_i)_{L_i} \; où \; E_i = \begin{cases}]a_i^0, a_i^1[\; si f_i \in \mathbb{R} \; (l'intervalle \; peut \; être \; fermé, \; semi-fermé, \ldots) \\ \{a_i^0, \ldots, a_i^1\} \; si \; f_i \in \mathbb{N} \\ \{0,1\} \; dans \; le \; cas \; binaire \end{cases} \tag{1.1}$$

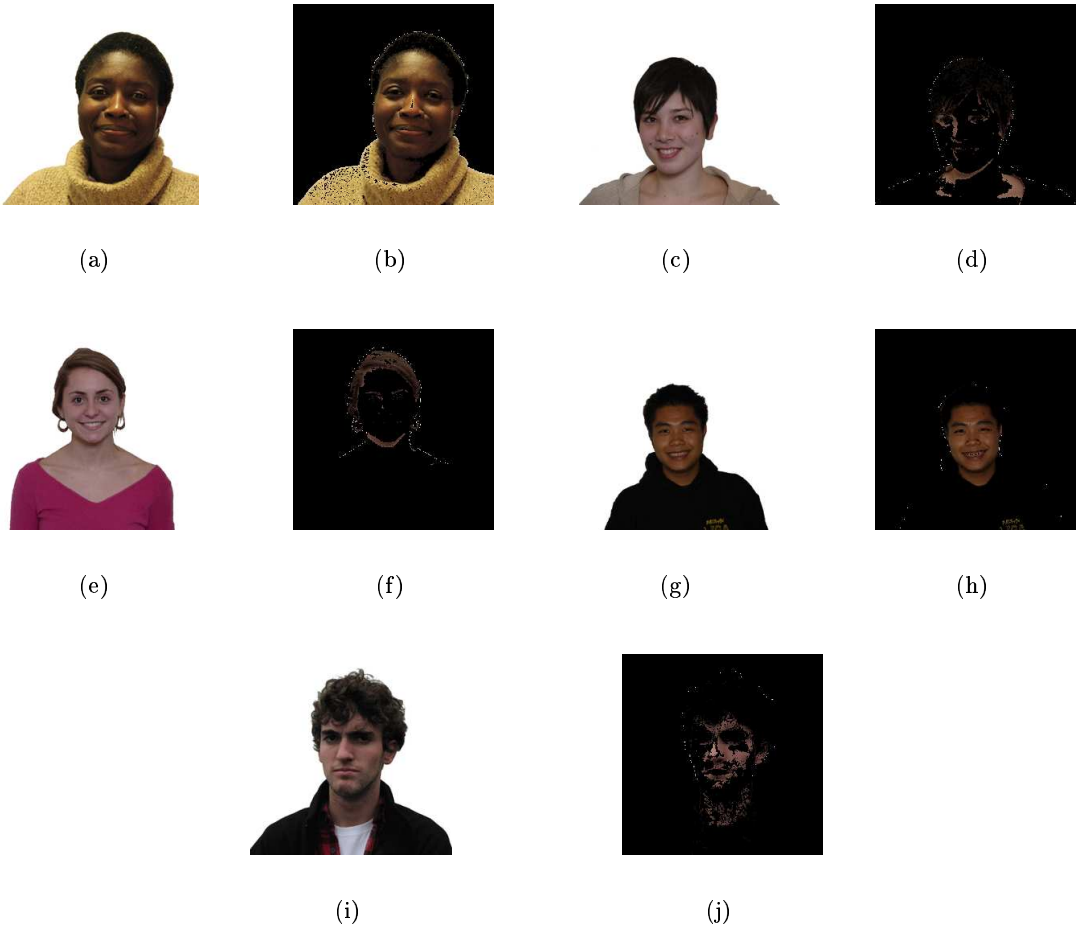
où les données f_i correspondent à des caractéristiques des objets (orientation, densité de contours, couleur, ...) et les données a_j correspondent à des valeurs numériques (entières ou réelles selon la nature de la caractéristique f_i à laquelle elles sont associées). Les caractéristiques f_i sont associées à un niveau de segmentation de l'image (pixel, région, ...) désigné par la donnée L_i.

Dans le cas de la détection des visages, il est possible de s'appuyer sur une caractérisation des pixels de la peau. Voici un exemple simple de modèle (tiré d'une partie de celui utilisé

dans [SP96]), appliqué à cet objet. Les pixels de la peau sont ici caractérisés selon un critère uniquement colorimétrique :

$$\mathcal{M}_{visages} = \{(H \in [0, 50]) \cap (S \in [0.23, 0.68])\}$$

où H et S désignent les composantes homonymes de l'espace HSL. L'utilisation de ce modèle aboutit alors aux résultats présentés dans la figure 1.5.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j)

FIG. 1.5 – Quelques résultats de la détection de visage, obtenus en appliquant le modèle *HSL* pour détecter la peau (les images a, c, e, g et i correspondent aux images initiales et les images b, d, f, h et j représentent les résultats obtenus).

L'analyse des résultats produits par le système montre que ceux-ci ne sont pas toujours suffisants : pour certains visages, les résultats présentent trop de fausses alarmes (cas du pull du visage a), dans d'autres cas, la proportion de pixels du visage qui n'ont pas été détectés est beaucoup trop importante (images d, f et j). Pour autant le résultat montré dans l'image h paraît suffisant : la zone de visage a été précisément extraite de l'ensemble de l'image. Le modèle de la peau choisi est donc imparfait : il ne convient pas à tous les visages . Cet exemple tend à invalider l'existence d'un modèle idéal permettant d'obtenir sur tous les types d'objets des performances parfaites. Nous verrons par la suite comme des considérations plus générales permettent de confirmer cette première conclusion.

1.5.3 Généricité des modèles et robustesse des systèmes

Lorsque les performances d'un système sont évaluées, deux critères doivent être pris en compte : la généricité du modèle induit de l'objet considéré, ainsi que la robustesse des opérateurs utilisés dans le système et dont l'objectif est justement de mettre en application ce modèle. Nous adoptons alors les deux définitions formelles de ces critères :

Définition 3. *Le degré de généricité d'un modèle se mesure au nombre d'objets étant des instances de celui-ci.*

Définition 4. *Le degré de robustesse d'un système se mesure en fonction du nombre de caractéristiques et de la largeur de la plage de valeurs de celles-ci pour lesquelles le système voit ses performances inchangées.*

Bien qu'il existe des passerelles entre elles, ces deux notions ne sont pas équivalentes. Supposons par exemple que le modèle ait un degré élevé de généricité. Selon la nature des opérateurs le constituant, le système n'est alors pas assuré d'être robuste. Si, par exemple, le degré d'illumination des objets n'entre pas dans la composition des caractéristiques du modèle et que, par contre, l'un des opérateurs du système voit son comportement altéré par des variations de cette caractéristique, le système n'est pas robuste.
Pour autant, des ponts existent entre la généricité du modèle et la robustesse du système qui lui est associé. Ainsi, il est commun de déclarer un système "*robuste à*" la variation d'une caractéristique particulière et ceci dans une plage de variation tolérée. Dans le cas de l'extraction d'objets, les caractéristiques suivantes peuvent être utilisées comme repères de la robustesse d'un système :

- l'orientation de l'objet dans l'espace
- la taille de l'objet
- le degré d'occlusion de l'objet
- la qualité de l'image et\ou de la vidéo
- ...

Il existe alors un lien entre la généricité du modèle et la robustesse du système si des caractéristiques communes sont utilisées dans la représentation de ces deux entités selon le mode proposé dans la formule 1.1.
La conception d'un système d'extraction d'objets performant s'appuie donc sur la définition d'un modèle générique auquel est associé un système robuste. Par la suite, la nécessité de l'adaptation sera abordée en prenant uniquement en compte la question de la définition du modèle et de sa généricité.

Si la nature de l'objet considéré et surtout son **cadre d'observation**, désignant l'ensemble des prises de vues possibles de l'objet (orientation, éclairage, ...) le permettent, il est possible de définir un modèle extrêmement précis de l'objet. Dans le cas d'une pièce mécanique observée sur une chaîne de montage par exemple, le **cadre d'observation** est connu. Il est donc possible de définir un modèle extrêmement précis de l'objet (par exemple un maillage 3D). Il n'en est pas de même pour des cadres beaucoup moins contraints, tels que les "visages" ou encore les "textes" apparaissant dans un document audiovisuel.
Dans le contexte de tels objets, le cadre d'observation est inconnu et la définition d'un modèle induit la construction d'un corpus de conception. La question est alors de savoir s'il est possible de construire des corpus de conception suffisamment larges pour contenir l'ensemble des instances différentes d'un même objet. Les figures 1.6 et 1.7 montrent alors la diversité des instances des objets "texte" et "visage".

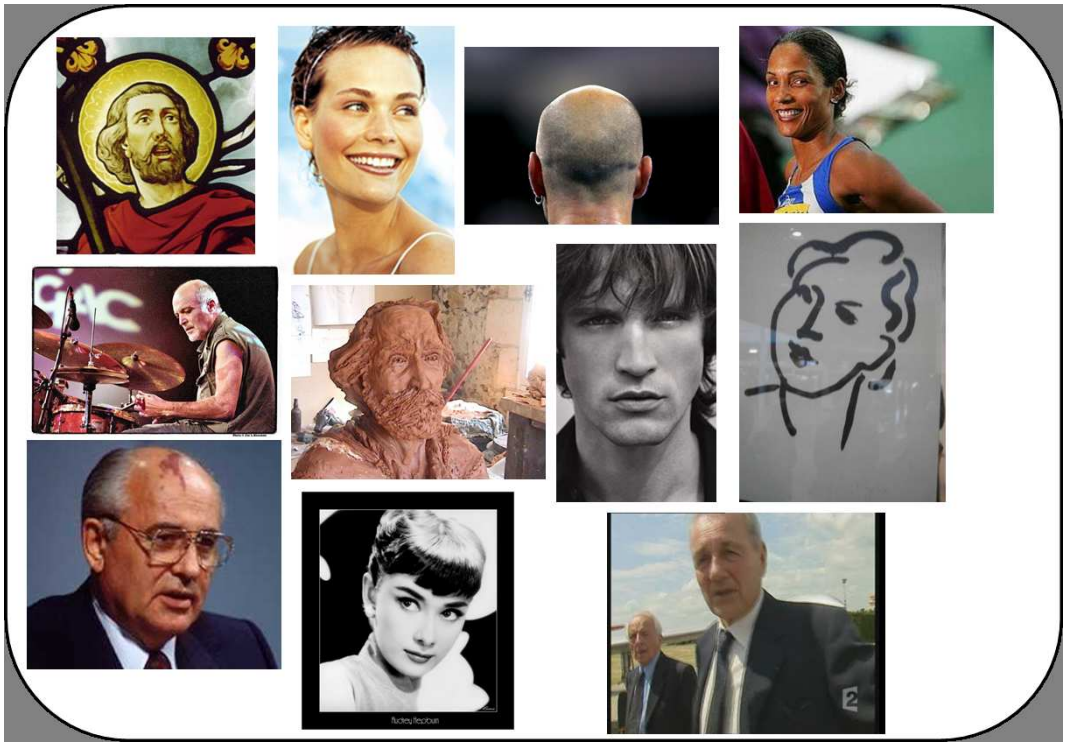


FIG. 1.6 – Différentes instances très différentes d'un même objet, l'objet "visage"

FIG. 1.7 – Différentes instances très différentes de l'objet "texte"

Les figures 1.6 et 1.7 montrent que la diversité des objets est telle qu'il est impossible de concevoir un corpus de conception à l'image de celle-ci. Le corpus de conception a pour principal objectif de permettre la délimitation des plages d'acceptation relatives aux différentes caractéristiques. Ainsi, si ce corpus est trop réduit, ces plages ne permettront pas de prendre en compte l'ensemble des instances différentes des objets pouvant être rencontrées dans les documents. La taille de ce corpus a donc une influence sur la quantité d'oublis produits par le système.
Dans le même temps, cette diversité soulève la question du choix des caractéristiques utilisées dans le modèle. Il apparaît alors clairement que la sélection de caractéristiques permettant de discriminer efficacement les zones d'une image correspondant à un objet des zones quelconques s'avère extrêmement difficile. La diversité a alors ici comme conséquence d'entraîner une sélection non optimale qui aura a priori des répercussions autant sur les oublis produits par le système que sur les fausses alarmes.

En conséquence, il apparaît que la définition d'un modèle idéal relève de l'utopie. Le concepteur du système est donc amené à effectuer un compromis : soit il choisit de satisfaire la contrainte de généricité auquel cas il s'expose à un grand nombre de fausses alarmes ; soit il préfère contraindre le modèle à une catégorie particulière d'objets, auquel cas le nombre d'oublis produits par le système augmente. Dans tous les cas, la taille du corpus de conception induit une contrainte quant à la nature des objets que le système sera capable d'extraire. En conséquence, que cette contrainte soit choisie explicitement ou non par le concepteur, elle n'en demeure pas moins effective du fait de la taille réduite du corpus de conception imposée par la diversité infinie des objets considérés.

1.5.4 Une unique solution pour maintenir les performances d'un système : l'adaptation

Nous venons de montrer que la diversité des objets empêchait le concepteur de construire un corpus de conception suffisamment large, ceci ayant pour principale conséquence de contraindre le modèle des objets sous-jacent. Les performances du système sur un corpus d'application quelconque CPS_{appli} sont alors dépendantes de ce que nous appelerons par la suite "distance orientée objet" entre deux documents (ici le corpus de conception et le corpus d'application), $D_0(CPS_{concept}, CPS_{appli})$. Cette distance correspond simplement à la distance entre les vecteurs de caractéristiques représentant les objets dans chacun des deux corpus considérés.
Dès lors il apparaît évident que la nature des performances du système sur le corpus CPS_{appli} est fonction de cette distance : plus celle-ci est importante, plus les performances seront dégradées. On pourra par ailleurs estimer que la capacité du système à maintenir un égal niveau de performances sur des corpus différents est fonction de la *généralité* du classifieur sous-jacent.

En conclusion, il apparaît que le maintien des performances d'un système d'extraction d'objets impose de modifier celui-ci, c'est à dire de l'**adapter**. Les modalités de ces modifications seront définies dans les deux chapitres suivants de cette première partie. Pour autant, il est possible de valider cette nécessité autour du système de détection de visages dont nous avons montré quelques résultats dans la figure 1.5. En effet, il apparaît que la modification des paramètres de ce système (à savoir les seuils concernant les caractéristiques colorimétriques utilisées) permettent d'obtenir de meilleurs résultats comme illustré dans la figure 1.8. En conclusion, la modification du système permet de résoudre le problème soulevé par la diversité des objets sur lequel le système est appliqué, diversité manifestée ici par la variation des teintes des différentes visages considérés.

(a) (b)

FIG. 1.8 – Deux résultats "meilleurs" (en termes d'oublis pour la première image et en terme de fausses alarmes pour la seconde) obtenus en appliquant deux nouveaux paramétrages du modèle (pour l'image b, le paramétrage est issu de [Lem03])

La figure 1.9 illustre finalement le point de vue adopté : nous considérons ici qu'un système associé à un modèle possède un espace de fonctionnement optimal (celui pour lequel les objets sont correctement extraits) qui ne se réduit pas au seul espace de fonctionnement originel. Nous considérons qu'une modification du système permet d'aboutir à de meilleurs performances relativement à un ensemble d'objets à extraire donné. L'objectif de notre étude est donc de *naviguer* intelligemment dans la zone de fonctionnement du système, cette navigation étant qualifiée par la suite d'**adaptation** du système.

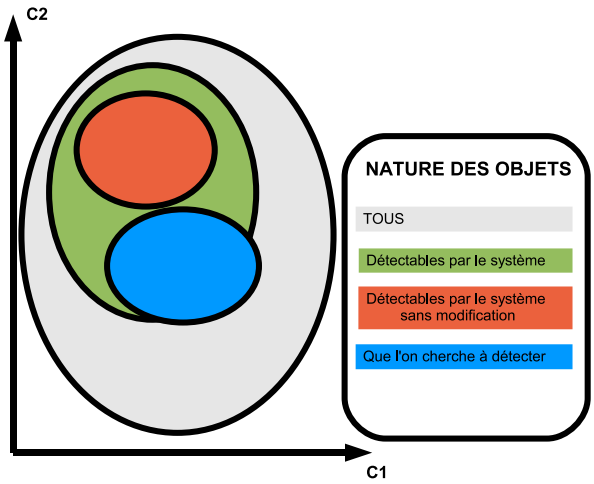


FIG. 1.9 – Une représentation simplifiée de l'espace des caractéristiques du modèle (réduit ici aux seules caractéristiques C1 et C2). L'ensemble des objets pouvant être détectés par le système (moyennant une adaptation de celui-ci) est délimité par la zone verte. Les zones rouges et bleues désignent quant à elles respectivement les objets pouvant être détectés par le système sans aucune modification, et les objets que l'on cherche à détecter. La non inclusion de cette seconde zone dans la zone des objets détectables par le système suggère la notion de **limitation intrasèque** au système sur laquelle nous reviendrons plus loin

1.6 Plan du manuscrit

L'ensemble des tâches recouvrant le vocable d'adaptation est particulièrement vaste et relève du domaine du **contrôle** des systèmes. Ces différentes tâches tout comme le domaine du contrôle en son ensemble seront détaillés dans le chapitre 2 de la partie 1, qui détaillera le processus de conception d'un algorithme de vision. Notre approche de l'adaptation, ainsi qu'un formalisme de représentation des systèmes de vision seront exposés dans le chapitre 3 de cette même partie. La méthodologie d'adaptation, qui constitue le coeur de notre travail sera présentée ensuite en deux chapitres, regroupés dans la partie 2 de ce manuscrit, chacun d'entre eux détaillant l'une des deux phases d'analyses mises en jeu dans notre méthodologie ; à savoir, l'analyse (indépendante du contexte d'application du système) du fonctionnement des différentes parties (ou modules) le constituant et l'analyse des résultats du système sur un corpus dit d'adaptation, noté CPS_{adapt}. Il sera alors démontré comment la mise en relation de ces deux analyses permet de pratiquer un diagnostic du système permettant de cibler la nature des modifications du système à effectuer pour en améliorer les performances.
Notre méthodologie d'adaptation est générique et peut s'appliquer à de nombreux objets différents. En gage de cette généricité, deux objets seront utilisés par la suite : l'objet "visage" nous permettra jusqu'à la fin de la partie 2 d'illustrer les choix effectués concernant la construction de la méthodologie. Pour autant , l'objet principal de cette étude demeure l'objet "texte". Nous présenterons ainsi dans la partie 3 l'ensemble des résultats produits par la méthodologie sur cette objet, relativement au système de détection de textes développé par Christian Wolf [Wol03]. Ces résultats seront détaillés dans le chapitre 2 de cette partie, tandis que le chapitre 1 aura pour enjeu de détailler les raisons nous ayant amené à priviliégier l'étude de cet objet particulier.
Enfin, une fois n'est pas coutume, des conclusions et de nombreuses perspectives à cette étude seront présentées dans une dernière partie.

2

L'adaptation des systèmes de vision : contexte scientifique, état de l'art et positionnement

Sommaire

Nous avons montré dans le chapitre précédent que le maintien des performances d'un système d'extraction d'objets sur un corpus d'application quelconque CPS_{appli}, dont les objets qu'il contient sont potentiellement différents de ceux contenus dans le corpus de conception $CPS_{concept}$, imposait de définir un mode de contrôle du système appelé adaptation. L'adaptation relève du domaine du contrôle des systèmes de vision et plus précisément du contrôle de son exécution.L'objet de cette partie sera alors principalement de présenter les différentes modalités de contrôle des systèmes de vision, nous permettant par la suite de positionner scientifiquement notre approche de l'adaptation.

Par ailleurs, nous présenterons dans un premier temps des contraintes plus générales liées à la conception des systèmes de vision, contraintes imposant justement la mise en place de systèmes de contrôle. Plus précisément, nous verrons comment la vue d'un système de vision en tant que résultat de la traduction opérative d'une volonté d'automatiser une tâche de compréhension visuelle, impose de pratiquer un ensemble de choix concernant la conception, choix justement facilités par l'utilisation de structures de contrôle adéquates.

2.1 Conception d'un système de vision, de la nécessité de mettre en place des structures de contrôle

Le principal particularisme scientifique des systèmes de vision repose sur leur mode de conception. Nous revenons ici sur ce trait en mettant l'accent sur les difficultés nécessairement rencontrées lors de la production d'un tel système. Ces difficultés seront subséquemment considérées comme la principale justification de l'émergence des structures de contrôle présentées dans la suite de cette partie.

2.1.1 Une multiplicité d'approches scientifiques

Définition 5. *Un système de vision a pour objectif de définir une formalisation (stochastique, numérique, analytique,...) d'**un** processus de la vision humaine.*

L'usage de l'article indéfini dans cette définition est volontaire : dans la vaste majorité des cas, l'objectif d'un système de vision est de se substituer à l'oeil humain dans le but de pratiquer une tâche restreinte à un contexte applicatif précis. En vue de simuler la séquence entière de traitements effectuée par l'homme pour accomplir une telle tâche de vision, un système de vision se doit idéalement de reproduire les phases de perception (et/ou d'acquisition) et celles de compréhension (et/ou de traitement) composant la chaîne humaine de transformations de l'information. Pour chacune de ces phases, une étape de modélisation puis d'approximation numérique des processus mis en jeu est alors nécessaire.
La réalisation informatique d'une tâche de vision requiert ainsi *a priori* des connaissances dans les domaines de la biologie, de la neuropsychologie, de la cognition, du traitement du signal ou encore de l'informatique. Une compréhension globale de la vision humaine, riche des approches proposées par ces différents domaines de recherche permettrait de mettre en place une simulation efficace de celle-ci. Pour autant, deux problèmes majeurs se posent : en premier lieu, les connaissances demeurent parcellaires, la vision humaine demeurant un thème de recherche particulièrement ouvert ; en second lieu, l'intégration des connaissances existantes s'avère extrêmement délicate du fait du cloisonnement scientifique relatif entre les différents domaines d'étude et de la difficulté à homogénéiser ces connaissances.
La pluridisciplinarité inhérente à la vision est donc un premier obstacle à la conception de systèmes de vision rigoureux. En sus de ces considérations, la question de la modélisation des images demeure elle aussi ouverte : de nombreuses théories coexistent ainsi (modèle continu, statistique ou numérique [Moi03]) et un même problème de vision peut ainsi aboutir à un ensemble de solutions différentes du point de vue du formalisme scientifique adopté.
Enfin, selon une approche systémique, tout système de vision peut être représenté comme une séquence d'opérateurs de traitements. Or le choix de ces opérateurs implique de prendre en compte un nombre extrêmement conséquent d'alternatives. Bien que reposant sur des principes souvent similaires, il est en effet courant qu'une même tâche (la détection de contours en constitue un exemple frappant) soit déclinée en de nombreuses versions. Le domaine de la vision souffre en effet de ce qu'il est une science entièrement expérimentale. Les procédés d'évaluation sont ainsi parfois particulièrement difficiles à mettre en oeuvre : évaluer les résultats d'algorithme de détection de contours est par exemple une tâche bien délicate. L'impossibilité de définir des bancs d'essais ainsi que des bases communes de test nécessite souvent aux équipes de recherches de rédevelopper leurs propres opérateurs, dont le fonctionnement est alors idéal pour leurs seuls usages.

En conclusion, il apparait que la conception d'un système de vision impose d'évoluer parmi différents domaines scientifiques, formalismes ou opérateurs. La première difficulté liée à l'agencement des briques d'un système de vision est donc de nature combinatoire : comment choisir parmi l'ensemble des possiblités, celle la plus à même de satisfaire les contraintes applicatives du problème étudié ?

2.1.2 Un problème mal défini

Une phase essentielle préalable à la réalisation d'un système de vision est de contraindre suffisamment le problème considéré pour que celui-ci admette une solution. En effet, il existe une importante différence entre les objets tels qu'ils sont appréhendés dans la formulation d'un problème de vision et les objets manipulés au cours du traitement. La phase d'acquisition des images entraîne nécessairement une perte d'informations (en passant d'une scène 3D à une représentation 2D une dimension est perdue) qu'il est difficile de recouvrer. A partir de cela, tout problème de vision est un problème mal-défini (ou sous-contraint) [Jol00] : les conditions initiales existantes sont insuffisantes pour permettre l'unicité des solutions. Dès lors, la formulation d'un problème de vision tout comme la traduction de ce problème en un plan d'actions nécessite la mise en oeuvre de contraintes sur les objets manipulés dans l'image, sur l'objectif du problème lui-même ou encore sur la nature des images (contenu et forme) utilisées. Une réflexion sur le système en son ensemble doit donc être menée.

David Marr fut le premier à définir un cadre précis à la conception des systèmes de vision. Selon lui, un système doit être appréhendé selon trois niveaux d'abstraction différents :

- *le niveau conceptuel*
- *le niveau algorithmique*
- *l'implémentation*

Etant donné un problème de vision, celui-ci doit être dans un premier temps énoncé clairement, c'est le niveau conceptuel. Les connaissances mobilisées ici sont principalement relatives au cadre applicatif envisagé et peuvent être exprimées dans un vocable propre à cette application. Le contexte de l'application, c'est à dire la nature des images devant être traitées doit aussi être défini. Un tel objectif peut être : *détecter et reconnaître les visages des présentateurs dans un flux audiovisuel de journal télévisé encodé au format MPEG2 à la résolution 750*400.* Le problème de vision doit ensuite être traduit en termes de traitements d'images. C'est l'étape de la **planification** qui correspond au niveau algorithmique. La notion de stratégie devient ici prégnante, il est question de choix, de justifications, de méthodes, d'ordonnancemment. Dans le cas de notre exemple, le système peut être appréhendé au niveau algorithmique de la façon suivante : *Segmenter le flux en plateaux/reportages, appliquer sur chaque image des séquences de plateau un détecteur de visage, suivre les apparitions d'un même visage dans le temps, reconnaître les visages en se basant sur la comparaison des zones extraites avec des signatures de visages conservées dans une base de donnée.* Vient finalement le niveau de l'implémentation. A ce niveau la séquence précise des algorithmes à utiliser (par exemple quel détecteur de visage) doit être établie et finalement, les informations concernant l'implémentation informatique de ces algorithmes est précisée : la nature des codes à exécuter, les liens vers les résultats, la valeur des paramètres, la syntaxe de la ligne de commande permettant l'exécution sont spécifiés. Cette étape se trouve énormément facilitée par l'utilisation d'une bibliothèque d'opérateurs [CEPR99, TCvdE94].
La paradigme de Marr constitue un apport essentiel à la méthodologie de conception des systèmes de vision. Il trace le cheminement général vers une solution. Pour autant il ne fournit pas

de méthodologie précise quant aux modalités de construction de systèmes effectifs. Si tenté que l'on applique précisément le paradigme de Marr, la question principale est alors de savoir comment acquérir puis gérer toutes les connaissances nécessaires, appartenant autant au domaine métier qu'à celui du traitement d'images, et ceci tout au long du cycle de vie du système. Par exemple, comment traduire les impératifs applicatifs du niveau conceptuel en un plan d'action algorithmique étant donné que les connaissances impliquées dans ces deux niveaux ne s'appuient pas nécessairement sur un formalisme commun ? En outre, combler les éventuelles lacunes en termes de connaissances demeure éminemment difficile.
La formalisation des connaissances tout comme la mise en place de procédures permettant de les combiner relève essentiellement du domaine de l'IA et sera abordé par la suite. Dans le cadre purement vision auquel nous nous limitons ici, l'interrogation la plus marquante porte sur l'élucidation des contraintes permettant de rendre un problème de vision soluble.
Une des premières solutions adoptée fut de considérer que la description de la scène observée rendait toute tâche de vision envisageable [AR91]. C'est le domaine de la *reconstruction de scènes*. Le postulat du reconstructionnisme est donc de réduire toute activité de vision à la seule tâche de description des scènes observées. L'obtention d'une telle description nécessite alors d'employer des contraintes excessivement fortes, telle que la contrainte de lissage, assimilant toutes les surfaces géométriques des scènes à des surfaces régulières. Initialement très usitée, cette contrainte montra rapidement ses limites ; les systèmes se trouvant tenus en échec dans le cas de scènes réelles pour lesquelles les conditions de prise de vue des objets invalident en règle générale la contrainte de lissage. Au reconstructionnisme s'oppose une approche beaucoup plus pragmatique de la vision, reposant sur la résolution de problèmes très spécifiques, pour lesquels de nombreuses connaissances *a priori* sont disponibles. C'est le cas en vision industrielle où, par exemple, un contrôle de qualité de pièces mécaniques est effectué en analysant des prises de vue de cette pièce. Ce contexte est extrêmement contraint : l'orientation de la prise de vue, l'illumination de celle-ci, tout comme une description extrêmement précise des objets rencontrés dans les images sont autant d'informations disponibles qui permettent de restreindre le problème initial (i.e : *Détecter les pièces défectueuses*) de sorte que les résultats obtenus soient satisfaisants. Dans le cas où des changements de conditions de prise de vue interviennent, les principes de la *vision active* et plus encore de la *perception active* mettent le capteur (ou l'observateur) au centre du processus de contrainte : lorsqu'une information est manquante, celui-ci est dirigé dans le but d'acquérir ces données par d'autres prises de vue. A la différence de l'approche de Marr qui considérait la scène comme l'objet à découvrir, la scène devient en vision industrielle une partie du système à concevoir. Il est paradoxal de noter que c'est en augmentant le nombre de degrés de liberté du système à concevoir, que des solutions, même particulières sont trouvées. Ce principe se retrouve par ailleurs dans les SVM[4], dont l'idée motrice est de projetter les données à classer dans un espace de grande dimension dans lequel celles-ci pourront être plus facilement partagées en différentes classes.

De la même façon que les systèmes de vision industrielle, la majorité des systèmes de vision s'orientent aujourd'hui vers une spécialisation très prononcée. Si la formulation du problème peut sembler floue de prime abord, par exemple *repérer les segments les plus importants d'un flux audiovisuel*, les contraintes choisies permettent de restreindre considérablement le champ de recherche du système. Le concept recherché, ici celui de *segment important*, est rapporté au niveau du signal, en définissant quelques quantités remarquables permettant de l'isoler dans le flux : couleurs vives, changements importants de la quantité de mouvement... Cette spécialisa-

[4]Support Vector Machines

tion applicative des systèmes entraîne donc d'une part une spécialisation des contraintes et des connaissances mises en oeuvres mais aussi une spécialisation des opérateurs qui les constituent. La navigation dans l'espace des opérateurs se trouve désormais conditionnée par les contraintes applicatives. La conception des systèmes s'en trouvent ainsi complexifiée et la notion de contrôle devient particulièrement pregnante.

2.2 Le contrôle des systèmes de vision

Le contrôle est défini dans [Gar00] comme les modalités de navigation d'un système de vision *"au sein d'un univers d'informations, de modèles, d'outils et de stratégies, en vue de résoudre le problème d'interprétation posé"*.

Nous adoptons alors par la suite une représentation simplifiée du contrôle, celui-ci étant envisagé à chacune des étapes du cycle de vie d'un système tel qu'illustré dans la figure 2.1 : lors de sa conception, de son exécution ou encore dans le cadre de sa réparation.

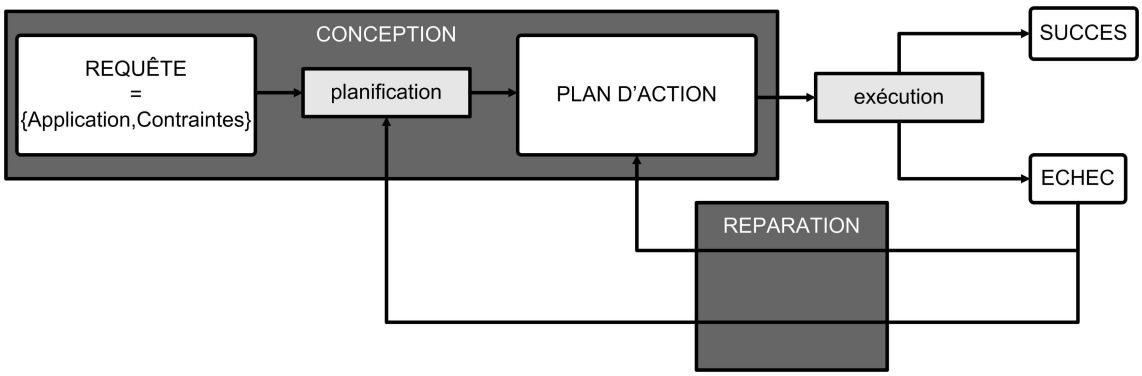


FIG. 2.1 – Vue générale (simplifiée) du cycle de vie d'un système de vision

Chacune de ces étapes sera détaillée dans la suite de cette partie. La question de leur automatisation et de leur contrôle sera, bien entendu, abordée avec un très grand soin.

2.2.1 La formulation du problème

La conception d'un système de vision fait appel à trois partenaires différents, possédant chacun une expertise et un savoir-faire spécifique. Ces trois partenaires sont l'expert du domaine applicatif, le traiteur d'images et le programmeur (cette dénomination s'inspire de celle proposée dans [Clo04] : client, concepteur, programmeur). Dès lors, trois représentations différentes d'un système de vision sont produites, permettant d'appréhender ce dernier aux trois niveaux d'abstraction proposés par Marr : en tant que requête du client, en tant que plan d'actions crée par le concepteur ou encore en tant que code exécutable implémenté par le programmeur. Première étape du cycle de vie d'un système, son expression sous forme de requête est source de nombreuses réflexions et la question du formalisme d'énonciation, que nous allons notamment développer ici, occupe une part non négligeable de ces réflexions.

Quelque soit le formalisme adopté, l'objectif de la requête initiant la construction d'un système de vision est invariablement le même : spécifier la nature de la tâche à effectuer ainsi que les contraintes lui étant spécifiques. Dans [Clo04], ces contraintes concernent la nature des

images à traiter ainsi que les critères de validation des performances du système. Quatre modèles représentent le système en son entier, tel que manipulé tout au long de son cycle de vie : le modèle du système, le modèle des images, le modèle des tâches et le modèle du programme. Les deux premiers modèles, ceux consacrés au système et aux images, portent les informations relatives à la formulation du problème ainsi qu'à l'expression des contraintes qui lui sont relatives.
Dans le modèle du système, ce dernier est appréhendé comme la partie centrale d'un système plus large comprenant une phase d'acquisition et une phase de post-traitement. Ce modèle décrit les objectifs de chacune de ces phases ainsi que du système étudié en propre. Les données permettant l'analyse des résultats sont elles-aussi développées :

- niveaux de détail,
- critères à optimiser,
- erreurs acceptables,
- éléments à inclure ou exclure,
- critères de performance,
- critères de qualité.

Ces informations du modèle système sont exprimées dans un vocabulaire qui "*emprunte à la fois à celui du domaine métier et celui du traitement d'images*" et sont stockées dans des formulaires.
Les images sont quant à elles décrites à trois niveaux. Au niveau physique, les images constituent le résultat d'un système d'acquisition. Au niveau perceptif, elles correspondent à un ensemble de caractéristiques de bas niveau (couleur, texture, ...). Enfin, au niveau sémantique, les images sont appréhendées comme compositions de différents objets. Le formalisme d'expression de ces informations diffèrent selon la nature du niveau considéré. On remarquera simplement qu'il est proposé de construire des diagrammes établissant les relations entres les différents objets constituant les images. Par ailleurs, le vocabulaire utilisé peut être plus contraint que celui manipulé pour établir le modèle du système.

L'expression de la définition du système (en tant qu'objectif de traitement sur des images particulières) peut être contraint (et dans le même temps facilité) par l'utilisation d'ontologies. Reprenant les travaux de Jean Charlet [Cha05] (prenant eux même en compte les réflexions proposées dans [Gru93, UG96]), les définitions suivantes d'une ontologie (relativement à un domaine particulier) sont adoptées :

Définition 6. *Ensemble des objets reconnus comme existant dans le domaine. Construire une ontologie c'est aussi décider de la manière d'être et d'exister des objets. (d'après [Cha05])*

Définition 7. *Une ontologie implique ou comprend une certaine vue du monde par rapport à un domaine donné. Cette vue est souvent conçue comme un ensemble de concepts -e.g. entités, attributs, processus-, leurs définitions et leurs interrelations. On appelle cela une conceptualisation... (d'après [Cha05])*

Les ontologies sont par exemple utilisées dans [RCR05], reprenant les préceptes de définition d'une application de traitements d'images énoncés dans [Clo04]. Deux ontologies sont créées. La première concerne les objectifs de transformation des images. La seconde, relative aux images repose sur les niveaux physiques, perceptifs et sémantiques précédemment cités. La tâche de traitement envisagée peut finalement être décrite en temps que concept métier, que primitive visuelle ou que partie d'une image.
Dans [MTB04, MTH04, HT03], un objectif plus spécifique est poursuivi : celui de la recherche d'objets dans les images. La formulation du problème consiste alors à modéliser la nature des

objets recherchés. Une ontologie, liée au domaine d'exploitation du système (par exemple les maladies des plantes dans [HT03]) permet d'effectuer cette tâche de modélisation haut-niveau. L'ontologie produite comporte alors les informations suivantes : caractéristiques spatio-temporelles, texturales et enfin colorimétriques des objets à extraire.

Pour faciliter la phase d'appariement entre les concepts du domaine énoncés par le client et le traitement de l'image, les objectifs peuvent être définis dans un formalisme intermédiaire entre celui du domaine applicatif et celui des opérateurs de traitement. C'est la solution proposée dans [DD98, JD96] où les objectifs sont énoncés en termes d'indices visuels. Trois langages de description entrent alors en jeu selon la hiérarchie présentée dans la figure 2.2. Selon ce

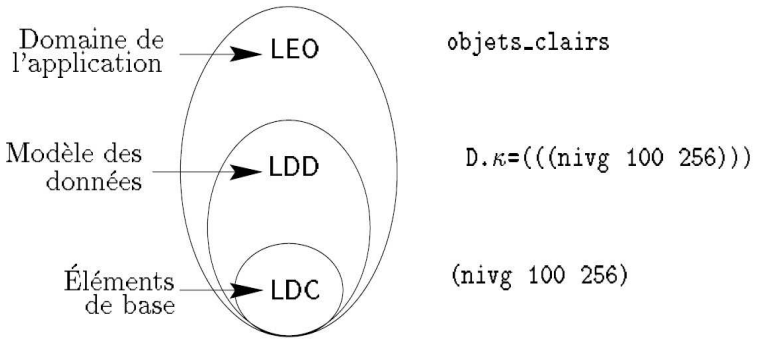


FIG. 2.2 – Hiérarchie de langage utilisée pour définir et traduire un objectif de traitement d'images : LEO correspond à un langage de bas-niveau d'expression des objectifs, LDD et LDC désignent respectivement un **L**angage de **D**escription des **D**onnées et un **L**angage de **D**escription de **C**oncepts (d'après [DD98])

formalisme, l'objectif est énoncé relativement aux images et aux objets qu'elles contiennent. Dans le système OCAPI [TCvdE94], qui s'appuie sur le paradigme du pilotage d'une bibliothèque d'opérateurs de traitement d'images, les objectifs sont définis en termes de traitements. La phase d'appariemment entre le haut niveau (les objectifs applicatifs et le bas niveau) se trouve ainsi éminemment simplifiée. Les informations retenues concernent :

- la nature des entrées et des sorties,
- des contraintes sur la qualité des résultats,
- des préconditions concernant les images en entrée et les résultats (postconditions),
- des règles d'évaluation,
- des critères de choix de la méthode.

Il est aussi envisageable de renoncer à l'emploi de ces formalismes souvent complexes. Dans le cas le plus extrême, la requête peut alors consister en un ensemble d'images annotées. C'est la solution retenue dans [DBB00].

L'énonciation du problème de vision à résoudre répond donc à une double contrainte : la proximité entre le formalisme adopté, celui du client et celui utilisé en aval lors de la phase de planification. Le choix d'un formalisme *équitable* est donc particulièrement important puisqu'il permet par la suite l'appariemment entre la requête et les opérateurs de traitement d'images composant le système. Il est à remarquer que les solutions retenues imposent en règle générale la constitution (souvent manuelle) de sources importantes de connaissances. Par ailleurs, l'appariemment entre les requêtes et les opérateurs repose sur un monde fermé, limité en autres choses

par la nature des concepts et/ou du vocabulaire utilisé. La capacité à évoluer de telles méthodologies paraît donc particulièrement précaire puisque l'ajout de nouveaux concepts impose une refonte du système global de construction des systèmes.

2.2.2 Du problème à la solution : la question de la planification

Les systèmes de vision sont profondément composites : la réalisation d'une tâche requiert d'appliquer en cascade un ensemble d'opérateurs de traitements différents. La difficulté à choisir ces opérateurs, très nombreux, a déjà été soulignée dans la partie 2.1. La phase de planification s'applique précisémment à cet ambitieux projet : proposer, étant donné un objectif énoncé selon l'un des formalismes précédemment cité, un plan d'actions (potentiellement hiérarchique), c'est à dire une séquence d'opérateurs à appliquer en vue de résoudre le problème posé. Selon la nature des informations disponibles, plusieurs solutions différentes ont été proposées.

Dans le cas de [DBB00], une méthode par apprentissage est utilisée. Le système de vision est modélisé comme un ensemble d'états successifs, les transitions entre ces états étant des actions, c'est à dire des procédures de traitement d'images. Le choix de la séquence optimale de ces actions repose alors sur la détermination de fonctions de récompense Q(s,a) où s désigne un état et a une action. Il est ici entendu que les opérations sont recompensées en fonction de leur capacité à faire évoluer le flux de données vers la forme attendue. Cette méthodologie s'appuie ici sur une représentation probabiliste des effets de chacun des opérateurs de la librarie. Une approche similaire est utilisée dans [DD98] où les effets des opérateurs sur les données sont au coeur du processus de planification.

Dans [TCvdE94] une première version du plan est connue. Le but est de contrôler son exécution dans un contexte particulier, c'est à dire en fonction des images présentées en entrée. Le plan contient initialement un certain nombre d'alternatives concernant la nature des opérateurs le composant (ces opérateurs sont déterminés *a priori*). Des critères de sélection sont attachés à ces branchements et sont évalués dynamiquement lors de l'exécution du plan. Une règle de production, dont la partie conditionnelle est relative au contexte d'application, c'est à dire à l'état des données traitées à l'instant où le choix doit être effectué, établit ainsi quel opérateur choisir.

Dans le système BORG [CEPR99], la phase de planification, hiérarchique, est basée sur un mécanisme de tableau noir [5]. Cinq niveaux d'abstraction sont considérés : la requête, l'objectif (quelles tâches pour répondre à la requête), le fonctionnel (quelles classes d'opérateurs pour effectuer les tâches (classification de pixels, etc)), la procédure (quelle méthode de traitement précise (binarisation par hystéresis)) et l'opérateur (le code de la librairie de traitement PANDORE utilisée [CPER95]). Pour construire un plan selon cette hiérarchie une base de connaissances ainsi qu'une base de données sont consacrées aux aspects de traitement d'images. Deux structures (ou sources) du même type sont dévolues au contrôle. Un planificateur intègre l'ensemble des connaissances des différentes bases pour proposer le plan. Le savoir en traitement d'images se décline en différentes décompositions de tâches, en des syntaxes d'exécution et des règles d'évaluation. La source liée au contrôle permet de choisir quelle type de connaissance en traitement utiliser. Une priorité des sources est ainsi définie. Le planificateur corrèle alors les informations

[5] Un tableau noir a pour objet de transmettre les informations entre les différentes sources de connaissances. Son fonctionnement est proprement asynchrone. A la différence d'un superviseur, il ne déclenche pas l'émission des informations par les sources. Le tableau noir constitue par ailleurs la référence de l'état courant du système.

pour construire le plan.

La plupart des systèmes présentés jusque ici mettent en avant la réutilisabilité des modules agencés pour des applications futures, potentiellement relatives à des requêtes différentes. Dans le cadre du **raisonnement par cas**, la phase de planification repose entièrement sur l'existence de plans effectifs qu'il convient d'adapter [FCPR98]. Nous reviendrons plus loin sur la question de l'adaptation de ces plans, nous contentant ici de développer la méthodologie de sélection initiale du plan le plus à même de remplir l'objectif envisagé. L'enjeu est donc de comparer les plans existants, modélisés sous la forme d'un duplet *<Problème de traitement d'images, Plan TMT de la solution à ce problème>*, le plan TMT se représentant lui-même en un triplet (Tâche (T), Méthode (M), Outil (T)) reprenant les trois niveaux d'abstraction de Marr. La distance entre deux cas, reprenant les principes de la distance d'édition, est basée sur le coût d'adaptation nécessaire pour que les deux cas considérés soient équivalents. Un ensemble de critères attachés au plan et à la définition du problème sont ainsi utilisés et la différence entre deux cas est alors établie à partir des différences entre ces critères. Ces derniers peuvent être de type numérique ou non. Dans le cas des critères non numériques (par exemple, les mots clés définissant la nature du problème de traitement d'images à résoudre) la différence correspond au nombre de termes en commun.

Dans les systèmes à base d'ontologies [MTB04, MTH04, HT03], l'enjeu consiste à associer l'ontologie du domaine applicatif, constituée de concepts visuels "métiers" avec un ensemble de traitements. Dans [MTB04], des détecteurs sont spécialisés lors d'une phase d'apprentissage en vue d'extraire les concepts contenus dans l'ontologie décrivant les objets qu'il convient d'extraire. Dans [HT03], deux niveaux de descriptions (stockées sous forme d'ontologies) sont utilisés ; un premier dit de "haut niveau" recensant les données dites "métiers" concernant les objets ; un autre dit de "bas niveau" décrivant les données "images". La traduction des premières données, abstraites, en données relatives au monde réel, celui des "capteurs", s'effectue par le biais d'un *système d'ancrage* (*anchoring system* en anglais). La recherche du plan permettant par la suite d'extraire ces données s'effectue selon des modalités similaires à celles exposées dans [TCvdE94], en pilotant une bibliothèque d'opérateurs.

Une nouvelle fois, il est important de remarquer le risque de sur-spécialisation des méthodes proposées qui, la plupart, s'appuient sur de nombreuses sources de connaissances dont la mise à jour s'avère complexe. Si le formalisme utilisé dans ces méthodes s'avère extrêmement utile du point de vue de la capitalisation des plans d'actions crées, des résultats ou de leur interprétation, les méthodologies de construction reposent essentiellement sur des connaissances *a priori* beaucoup trop figées pour espérer permettre une extension de leur utilisation à des problèmes de vision autres que les problèmes de traitement relativement simples sur lesquels ces méthodologies sont expérimentées. Le goulet d'étranglement de ces méthodes demeure donc essentiellement la phase, inévitable, d'acquisition des connaissances, inconvénient par ailleurs reconnu par la communauté [DH96].

En marge des méthodologies de planification présentées ici, il convient de rappeler à titre informatif, l'existence d'outils dédiés à la constitution de plans d'actions reposant sur la programmation par flot de données : les systèmes sont conçus en reliant entre elles des boîtes noires effectuant des traitements sur l'image. Parmi les plus *célèbres*, on citera notamment Khoros et

l'IHM [6] associée Cantata [KR94, AFK01], ou encore, dans un domaine différent Simulink [7].

2.2.3 Le contrôle de l'exécution

La validation des résultats de la planification repose essentiellement sur la justesse des connaissances *a priori* sur lesquelles s'appuie la production du plan d'actions. Il n'existe donc aucune assurance de la qualité de ces résultats et une phase de contrôle de ceux-ci s'avère nécessaire. Le contrôle de l'exécution consiste alors en deux étapes distinctes : d'une part l'évaluation des résultats du système conformément aux contraintes de performances établies lors de la formulation du problème ; d'autre part la mise en oeuvre de processus de modification du système (on parle de réparation) en vue de corriger les erreurs constatées. L'automatisation du contrôle de l'exécution tend ainsi à extraire le concepteur du cycle manuel "*essai-correction*" nécessaire à la finalisation de son système. Il est à remarquer ici que c'est l'absence d'un formalisme analytique des opérateurs qui impose un tel mode de fonctionnement aux concepteurs : bien que certaines tendances globales de comportement relativement aux données d'entrée soient connues, il n'existe pas de solution permettant de prédire la qualité des résultats d'un opérateur en fonction de ses entrées.
Deux *instants* du contrôle de l'exécution doivent ici être distingués :

1. le temps de l'**instanciation**, durant lequel le système est modifié en vue d'obtenir les performances attendues sur le corpus de conception $CPS_{concept}$
2. le temps de l'application, où l'objectif est généralement d'ajuster le système en vue de prendre en compte des changements pouvant toucher :
   - aux contraintes de performances
   - à la nature des données d'entrée à traiter

Dans tous les cas, en ce qui concerne la nature des modifications éventuelles proposées par le système de contrôle, celle-ci dépend de la méthodologie employée : autant ces modifications peuvent se limiter aux seuls paramètres des opérateurs constituant le plan, autant certaines méthodes envisagent des modifications bien plus profondes, remettant en cause la validité des choix effectués lors de la planification (on parlera ici respectivement d'*optimisation* ou de *re-planification*).

Dans [SMV+98], les systèmes sont considérés comme étant capables de s'auto-ajuster au contexte. Cet auto-ajustement (*self-tuning*) repose sur la définition d'une part de règles d'évaluation et d'autre part de règles d'ajustement. Le champ des modifications envisagées embrasse l'optimisation des paramètres tout comme la re-planification. Deux modes de contrôle sont distingués : dans le *mode utilisateur*, l'évaluation des résultats se limite aux résultats finaux ; dans le *mode spécialiste* les résultats intermédiaires de la chaîne de traitements sont supposés disponibles et l'évaluation porte alors sur chacune des étapes de celle-ci. Enfin, deux types d'évaluation sont envisagés. Selon le degré de précision de l'estimation des performances, celle-ci peut être qualifiée de générale ("*trop de fausses alarmes*") ou de spécifique ("*cet objet est une fausse alarme*"). Quelque soit le type d'évaluation considéré, celle-ci repose systématiquement sur l'intervention d'un expert visualisant les résultats. Coûteuse dans les deux cas, l'évaluation, lorsqu'elle est spécifique, nécessite de surcroît de mettre en place des IHM particulièrement complexes. L'évaluation *générale* lui est donc préférée dans cette étude. La sélection de la modification à apporter au système en cas d'échec est conditionnée par un ensemble de règles de production. L'objectif

[6] **I**nterface **H**omme **M**achine
[7] `www.mathworks.com`

essentiel de ces règles est de parvenir à restreindre *a priori* le champ des modifications possibles, notamment en ce qui concerne l'optimisation pouvant mettre en jeu un espace de recherche particulièrement important lorsque les paramètres sont nombreux et possèdent une plage de variation étendue. Par ailleurs, certaines de ces règles de production permettent la transmission de l'évaluation à des opérateurs précédents ou de niveau supérieur dans la hiérarchie [MVvdEvH95].

Dans le système BORG [CEPR99], les sources de connaissances utilisées (comme, par exemple, la décomposition d'une tâche) sont qualifiées selon leur aptitude à résoudre un certain type de problème dans un certain contexte. Ces taux d'aptitude sont utilisés comme mesure de cohérence du système et selon leur évaluation il est possible de considérer que le système a atteint une impasse. Dans cette situation, la construction du plan reprend depuis le niveau d'abstraction supérieur. Un autre mode d'évaluation entre en jeu dans le processus de contrôle : à chaque décomposition est attachée une règle d'évaluation. Si tous les critères d'évaluation des sous-tâches de chaque décomposition sont remplis, l'exécution est un succès. Au contraire, dès que ce critère n'est pas rempli par une des sous-tâches, la décomposition entière est supprimée et une phase de re-planification est exécutée.

Dans le cadre du raisonnement par cas [FCPR98], l'adaptation consiste à affiner le plan d'action choisi initialement par proximité avec l'application envisagée. Cette adaptation consiste à rechercher récursivement, pour les sous-tâches du plan qui ne satisfont pas les critères retenus, d'autres décompositions plus adaptées dans la base de celles disponibles.

Pour restreindre la dépendance aux connaissances *a priori* nécessaires, par exemple, à l'énoncé de règles de production, certains systèmes (que nous qualifierons de systèmes "*autonomes*") reposent sur une méthodologie "intelligente" d'optimisation. C'est le cas par exemple dans [PBC97, TDR04], où l'optimisation est limitée aux seuls paramètres auxquels le système est estimé être sensible (paramètres **influents**). Cette analyse de la sensibilité se substitue ici aux règles de production. Par ailleurs, dans [TDR04], un plan d'expériences permet de réduire encore, lors de l'optimisation, le nombre de combinaisons à tester en vue de trouver le paramétrage optimal. L'objet de ces méthodes est donc essentiellement combinatoire : réduire la taille de l'espace des paramétrages à parcourir en vue de déterminer le paramétrage optimal.

2.3 Conclusion

A l'issue de cette étude, tant les considérations générales sur le domaine de la vision que l'état de l'art [8] relatif au contrôle des systèmes de vision, mettent en lumière tous les obstacles se dressant devant la conception d'un système de vision et plus encore, en ce qui nous concerne, devant la construction d'un système de contrôle effectif.
En particulier, au delà de la difficulté existant à acquérir les connaissances nécessaires au bon fonctionnement des systèmes, l'appariemment entre les connaissances expertes "haut-niveau" et les connaissances liées au traitement, dites de "bas-niveau", repose sur une modélisation manuelle et intuitive des liens de causalité entre ces deux sources. La subjectivité, tout comme la non évolutivité d'une telle approche suscitent des interrogations justifiées quant à sa validité. Une solution consiste alors à se départir de telles limitations en restreignant l'incidence

[8] Cet état de l'art ne se veut pas exhaustif et tend plus à donner une image de la recherche en contrôle des systèmes de vision pour positionner nos travaux, qu'à en proposer une analyse circonstanciée. Pour une telle étude, la lecture de l'analyse proposée dans [Gar00] s'avère indispensable.

des connaissances extérieures dans le fonctionnement des méthodologies de contrôle “à base de connaissances”.
Le prochain chapitre présentera ainsi notre méthodologie d'adaptation des systèmes de vision, fondée sur le principe d'*autonomie du contrôle*, consistant à circonscrire autant que possible le volume et la nature des connaissances *a priori* nécessaires à son fonctionnement. Nous verrons comment le système, considéré comme une source propre de connaissances, peut suppléer aux sources externes utilisées habituellement.
Plus précisément, l'adaptation relève de la tâche du contrôle de l'exécution du système, et encore plus spécifiquement dans notre cas, de l'instant de l'application, lorsque la nature des données en entrée varie. Nous avons vu que cette tâche consistait essentiellement à choisir la nature des modifications à effectuer sur le système pour optimiser ces performances. Ce choix repose, lorsque le contrôle est dit "*à base de connaissances*", sur l'utilisation de connaissances permettant de modéliser l'association entre l'analyse des performances et la nature des modifications à effectuer. Dans le cas des systèmes "*autonomes*" présentés, la nature des modifications relève de l'analyse de l'influence des différents paramètres. Nous avons déjà présenté les limitations des systèmes "*à base de connaissances*". Concernant les systèmes "*autonomes*" existants, la principale limitation relève du caractère "*aveugle*" de l'optimisation proposée : si il existe un réel ciblage de l'optimisation aux paramètres influents, il n'en demeure pas moins qu'il n'existe pas d'analyse du lien entre le caractère influent d'un paramètre et la notion de **responsabilité** (de l'erreur constatée) de l'opérateur de traitement auquel celui-ci est attaché.
En sus de l'autonomie, la notion de **responsabilité** est tout aussi centrale dans notre méthodologie d'adaptation et nous verrons ainsi par la suite comment cette dernière s'articule autour de ces deux notions.

3

Proposition d'une méthodologie d'adaptation

Sommaire

L'étude des méthodes de contrôle des systèmes de vision, et plus spécifiquement des méthodes liées au contrôle de l'exécution de ces systèmes a fait apparaître dans le chapitre précédent un ensemble de limitations auxquelles la méthodologie d'adaptation détaillée dans cette étude tente d'apporter une réponse.

Nous avons vu que cette méthodologie reposait sur deux notions : d'une part celle d'**autonomie** consistant à limiter l'utilisation de connaissances *a priori* sur le mode de fonctionnement du système considéré ; et d'autre part la notion de **responsabilité**, attachée à la découverte, pour une erreur donnée, du module du système responsable. L'enjeu de cette partie est ainsi de montrer comment notre méthodologie d'adaptation des systèmes d'extraction d'objets prend en compte ces deux contraintes.

Dans un premier temps, la nature des modifications du système permissibles doit être établie. Ceci consiste à choisir entre optimisation, re-planification ou les deux modes utilisés conjointement, et il sera alors montré que la contrainte d'autonomie impose le choix unique de l'optimisation. Par la suite, l'objectif de la méthodologie d'adaptation sera établi précisément, avant

de détailler les contraintes liées à son application. Ce chapitre se terminera alors par la présentation de l'organisation de la méthodologie dont nous verrons qu'elle repose sur deux analyses distinctes : la première analyse, dite "analyse des comportements" du système, qui relève de l'évaluation des performances de celui-ci sur un corpus d'adaptation et la seconde, dite "diagnostic de responsabilité, qui a pour objectif de mettre en relation une analyse du fonctionnement du système indépendante du corpus d'adaptation avec les résultats produits par l'"analyse des comportements".

3.1 Choix du mode d'adaptation

3.1.1 Réduction des connaissances disponibles

Le contrôle des systèmes de vision exige l'utilisation d'une quantité non négligeable de connaissances. Le principe d'autonomie entend alors réduire la participation de telles sources externes dans la phase de capitalisation nécessaire au contrôle. Nous revenons ici sur les différents types de connaissances nécessaires *a priori*. La nature des connaissances dont nous ne supposons pas la disponibilité dans le cadre de notre méthodologie d'adaptation sera par la suite exposée.

En référence à l'état de l'art, la taxonomie suivante des connaissances nécessaires est adoptée

- *Connaissances sur le problème à résoudre* : nature de l'objet à extraire (visage, texte, ...), objectifs applicatifs envisagés (indexation, segmentation spatio-temporelle, ...), conditions d'acquisition des documents (format des images, encodage de la vidéo, ...), règle d'évaluation globale du système.
- *Connaissances sur le plan d'actions* :
  - composition du plan : intitulé des décompositions, nature des opérateurs élémentaires utilisés (syntaxe d'exécution, paramétrage, ...),
  - justifications des décompositions adoptées.
- *Connaissances pour le contrôle de l'exécution* : règles d'ajustement du système, régles d'évaluation des modules de chaque décomposition.

Sont supposées inconnues dans cette étude, les connaissances relatives à la justification du plan et celles concernant le contrôle de l'exécution (en particulier les règles d'ajustement du système). Par ailleurs, il est aussi admis qu'aucune description formalisée des effets des différents modules composant le système n'est connue. En conséquence, l'autonomie entraine la réduction des connaissances disponibles aux seules entrées/sorties du système. Si les entrées correspondent simplement aux flux vidéo sur lesquels sont appliquées les systèmes d'extraction d'objets que l'on cherche à adapter, la nature des sorties disponibles des systèmes dépend de la formalisation adoptée de ceux-ci et sera détaillée par la suite.

3.1.2 L'optimisation imposée par la contrainte d'autonomie

Nous avons vu dans le chapitre précédent que deux types de modifications d'un système pouvaient être envisagées pour permettre son adaptation : l'optimisation, consistant à ajuster la valeur des paramètres et la re-planification autorisant la modification de la nature des modules composant le système. La contrainte d'autonomie impose alors de réduire le champ des modifications possibles à la seule optimisation et ceci pour deux raisons :

- **le coût élevé de la re-planification** : la re-planification consiste à remettre en cause la construction du système. Certains modules ou séquences de modules peuvent ainsi être remplacés par des modules ou séquences équivalentes. Le coût de la re-planification repose alors sur la conception et le maintien d'une base de modules permettant d'effectuer de telles substitutions. Par ailleurs, la possibilité d'effectuer ces opérations impose l'existence d'un framework de programmation particulièrement efficace dont le coût de développement est loin d'être nul, notamment lorsque les séquences à construire sont non-homogènes du point de vue des types des données échangées au long de celles-ci.
- **la nature des connaissances disponibles** : sans une connaissance pointue du fonctionnement du système et des modules le composant, le déclenchement de la re-planification impose, dans un premier temps, de tester l'ensemble des paramétrages possibles de chaque opérateur en vue de découvrir si les performances peuvent être améliorées sans pour autant substituer un module à un autre, opération coûteuse pour les raisons précedemment citées. L'optimisation apparaît alors comme une étape nécessaire préalablement à la re-planification lorsque les connaissances disponibles sont réduites. Enfin, laisser le champ libre à la re-planification soulève la question de savoir différencier l'ajustement de la création d'un nouveau système.

L'optimisation propose ainsi un mode de modification moins coûteux, et constitue un mode incontournable lorsque les connaissances disponibles sont réduites. En conséquence, l'optimisation s'accorde pleinement avec la contrainte d'autonomie adoptée.

3.2 Objectif de la méthodologie

L'objectif principal de la méthodologie est le suivant : **"Permettre de cibler l'optimisation des paramètres d'un système d'extraction d'objets relativement à un corpus de document particuliers (le corpus d'application CPS_{appli}), aux seuls paramètres des modules le constituant, responsables des différents types d'erreurs observés".** Cet objectif rend donc compte de la difficulté de concevoir un modèle générique des objets et propose de cibler la nature des ajustements à effectuer lorsqu'un système d'extraction d'objets est appliqué sur CPS_{appli} dont les objets diffèrent de ceux contenus dans le corpus de conception.

Par ailleurs, cet objectif sous-entend l'organisation de notre méthodologie autour des deux analyses déjà mentionnées : l'*analyse des comportements*, permettant d'extraire les différents types d'erreurs, et le *diagnostic de responsabilité* permettant de cibler les modules responsables ; analyses dont les rouages principaux seront présentés par la suite.

La réalisation de cet objectif aura alors deux principales conséquences :

1. **Un gain combinatoire** : certains systèmes d'extraction d'objets peuvent mettre en jeu un nombre important de paramètres ayant chacun de larges plages de variation. La focalisation de l'optimisation aux seuls paramètres des modules responsables aura ainsi pour conséquence de réduire la complexité de la recherche du paramétrage optimal,
2. **La compréhension du système** : le principe d'autonomie nous impose une découverte automatique du mode de fonctionnement des systèmes, et ceci en exploitant uniquement leurs entrées/sorties. Notre méthodologie propose donc une tentative de compréhension automatique des systèmes d'extraction d'objets.

3.3 Contraintes et pré-requis de la méthodologie

L'adaptation met en jeu un système d'extraction d'objets que l'on souhaite appliquer sur un corpus d'application CPS_{appli}. L'objet de cette partie est alors de préciser quels sont les contraintes relatives aux systèmes considérés, aux objets pris en compte ainsi qu'à la définition d'un corpus d'adaptation CPS_{adapt}, relativement au corpus CPS_{appli} concerné.
Nous verrons ainsi par la suite quelle représentation des systèmes de vision et plus précisément des systèmes d'extraction d'objets est adoptée en soulignant la nature des sorties de ceux-ci disponibles pour mettre en oeuvre notre méthodologie. Il sera par ailleurs montré comment la représentation des systèmes d'extraction d'objets adoptée impose le choix d'une adaptation séquentielle de ces systèmes. La restriction à une classe d'objet particulière ainsi que la définition de CPS_{adapt}, liée à la prise en compte de contraintes applicatives seront par la suite détaillées.

3.3.1 Représentation des systèmes

3.3.1.1 Cas général des systèmes de vision

Représentation hiérarchique Le paradigme de Marr évoqué dans la partie 2 constitue un point d'ancrage commun à de nombreuses représentations des systèmes en général et des systèmes de vision en particulier. L'approche retenue dans cette étude s'appuie elle aussi sur cette vue hiérarchique des systèmes. Inspirée des préconisations définies dans [Clo04], la représentation adoptée est illustrée dans la figure 3.1.

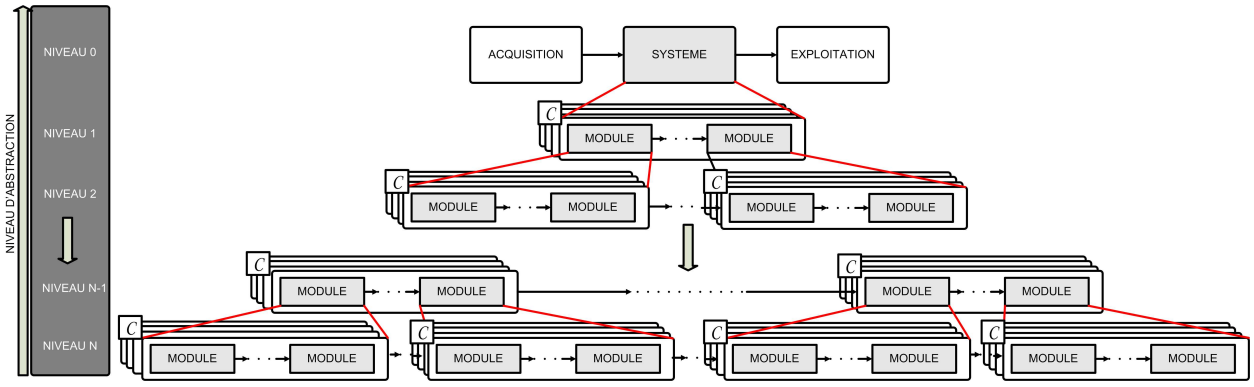


FIG. 3.1 – Représentation hiérarchique des systèmes de vision

A la différence de l'approche proposée dans [Clo04], aucune restriction n'est imposée quant au nombre de niveaux de la hiérarchie. En effet, il est ici admis que la profondeur idéale est fonction de la simple possibilité à décomposer un module en une séquence d'autres modules. Il existe cependant une limitation à cette représentation : considérant le dernier niveau de la hiérarchie, cette dernière est dite complète si les modules la composant sont de type élémentaire.

Définition 8. *Un module est dit* ***élémentaire*** *si l'ensemble de ses décompositions possibles relèvent de son implémentation.*

A titre d'exemple, le gradient de Sobel ou une érosion morphologique utilisant un masque 3x3 centré sont des modules élémentaires (on retrouve ici simplement le niveau algorithmique proposé par Marr).
Il est possible que le plan d'actions permette de choisir entre plusieurs possibilités au cours de son application, en fonction de critères définis *a priori* (ces critères peuvent porter par exemple

sur la nature des résultats obtenus). Ces possibilités peuvent concerner un unique module ou une décomposition entière. La représentation du système ne se trouve alors pas modifiée. Un critère $\mathcal{C}$ (tel que noté dans la figure 3.1) est dans ce cas attaché au module ou à la décomposition concerné (pour des raisons de lisibilité, seules les alternatives concernant les décompositions sont représentées dans la figure 3.1).
Dans le cas du traitement d'images, il est entendu que le flot des données émises entre les différents modules est homogène (de type "*image*"). Dans le cas plus général des systèmes de vision, ce critère d'homogénéité peut ne pas être respecté. Dans ce cas, une décomposition est déclarée "licite" si le flot de données observé est cohérent, c'est à dire si l'entrée des modules est du même type que la sortie des modules les précédent dans la séquence.

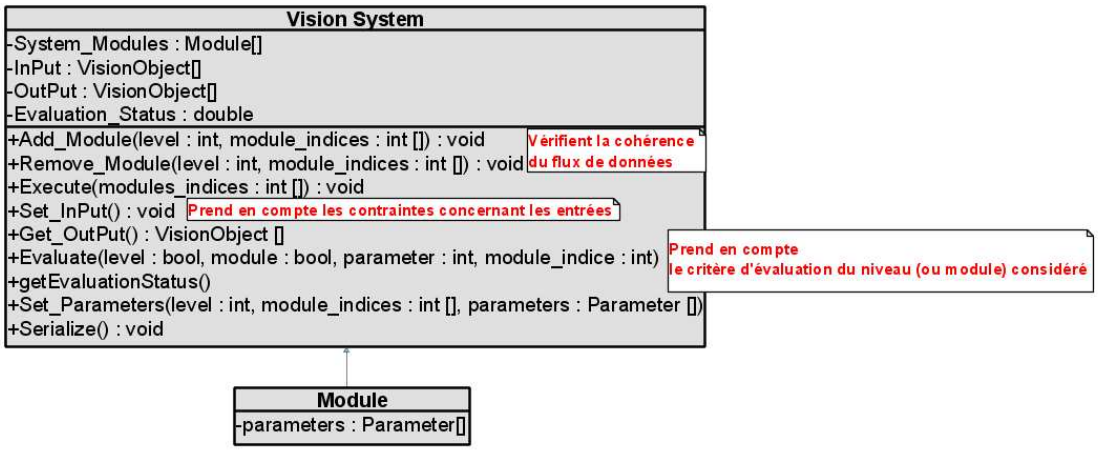
Représentation informatique Dans le cadre général de l'utilisation des systèmes ainsi que dans celui, particulier de l'adaptation, le formalisme objet constitue une solution particulièrement adaptée à la représentation des systèmes. En marge des procédés d'instanciation et d'exécution parcellaire (seulement un certain module ou une certaine décomposition) du plan, la sérialisation des objets constitue un atout indéniable dans le cadre de la conservation des résultats (pour conserver par exemple les informations relatives au paramétrage et aux performances associées). Une représentation UML des sytèmes de vision est ainsi présentée dans la figure 3.2.

3.3.2 Le cas particulier des systèmes d'extraction d'objets dans les flux audiovisuels

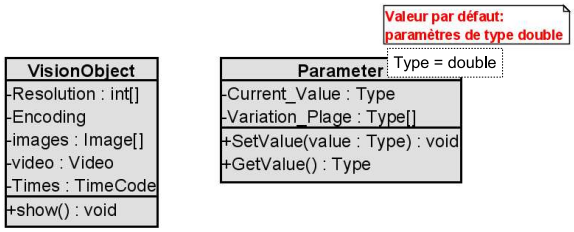
Nous avons déjà énoncé dans le premier chapitre de cette partie quelques unes des particularités des systèmes d'extraction d'objets (appelés systèmes DSRO par la suite [9]), notamment la décomposition de ceux-ci en 4 modules : la détection, le suivi, l'amélioration et la reconnaissance. Chacun de ces modules joue un rôle bien précis déterminé ici par la nature de ses sorties :

1. **Détection** : associe à chaque image un ensemble de boîtes englobantes délimitant l'ensemble des objets détectés (ou des images dans lesquelles les objets sont isolés). Deux méthodologies sont envisageables : étudier effectivement chaque image, ou déterminer toutes les positions des objets par interpolation, relativement aux positions de ceux-ci lors de leur apparition et de leur disparition.
2. **Suivi** : associe les différentes instances temporelles d'un même objet. La sortie du module de suivi consiste donc en un ensemble d'objets spatio-temporels. Une nouvelle fois, deux méthodes se différencient. Dans les deux cas, dès qu'un objet est détecté, un *suiveur* (*tracker* en anglais) dont l'objectif est de repérer ce même objet dans l'image suivante lui est associé. Deux types de fonctionnement de ce suiveur sont possibles :
   (a) associer l'objet O auquel il est attaché à un objet particulier détecté dans l'image suivante par comparaison des caractéristiques de l'objet O avec celles de l'ensemble des objets détectés dans l'image suivante.
   (b) prévoir le déplacement de l'objet (par l'utilisation de filtres de Kalman par exemple) et proposer une délimitation dans l'image suivante sans effectuer la phase de détection.
3. **Amélioration** : le module d'amélioration est facultatif. Son objectif est de préparer les objets à leur reconnaissance. Cette préparation consiste habituellement à isoler des zones particulières des objets, à augmenter leur résolution, ou encore comme nous le verrons dans le cas des textes, à réduire la complexité du fond sur lequel ceux-ci sont écrits en

[9] Détection, Suivi et Reconnaissance d'Objets



(a) Description de la classe représentant un système de vision



(b) Description des classes représentant les objets manipulés par un système de vision : ses entrées/sorties (VisionObject) ainsi que ses paramètres

FIG. 3.2 – Schéma UML adopté pour la représentation objet des systèmes de vision. L'ensemble du modèle facilite l'application et l'évaluation parcellaire des systèmes : chaque module ou séquence de modules peut être exécuté et évalué.

vue de faciliter la binarisation. La sortie du module d'amélioration consiste en un ensemble d'images. Dans le cas où la méthode utilisée s'appuie sur l'intégration temporelle des différentes instances d'un même objet, une unique image est associée à chaque objet. Dans le cas contraire, la sortie du module d'amélioration contient autant d'images que la sortie du module de suivi en propose (c'est à dire autant d'instances différentes d'un même objet associées lors du suivi).

4. **Reconnaissance** : la reconnaissance consiste à associer à l'objet un texte ou un autre objet (par exemple associer un visage à une autre prise de vue de celui-ci). Dans le cas de la reconnaissance "*textuelle*", l'identification proposée représentera, par exemple, le nom de la personne pour l'objet *visage* ou la transcription du texte dans le cas de l'objet *texte*.

En conséquence, la représentation des systèmes DSRO repose sur une hiérarchie de profondeur supérieure ou égale à 3 : le niveau du système, celui des 4 modules qui viennent d'être présentés et enfin celui des modules élémentaires (un détecteur de contours de Canny entrant

dans la composition du module de détection par exemple) constituent des niveaux indispensables à une représentation cohérente. Un, ou plusieurs niveaux supplémentaires peuvent être ajoutés à cette représentation le cas échéant selon le mode de construction du système considéré.

3.3.3 Mode d'adaptation retenu

Adapter un système DSRO consiste, lorsque notre méthodologie est appliquée, à pratiquer les deux analyses déjà mentionnées relativement aux résultats produits par le système sur le corpus d'adaptation. Si le système est considéré en son ensemble, sa sortie finale correspond à celle du module de reconnaissance et les performances sont donc analysées relativement à ces résultats. Pour autant ce mode d'adaptation n'est pas satisfaisant. En effet, selon le contexte d'utilisation du système (indexation, segmentation, ...), il peut être intéressant d'obtenir des informations sur les performances des modules de détection, suivi, ... Si l'adaptation repose uniquement sur l'évaluation des résultats de la reconnaissance, un tel retour devient impossible. En effet, selon leur mode de fonctionnement, il est envisageable que le système obtienne des résultats globaux (c'est à dire à l'issue de la phase de reconnaissance) satisfaisants, sans que les résultats des phases intermédiaires ne le soit. En effet, les critères d'évaluation diffèrent selon la nature des modules considérés. Concernant l'extraction d'un visage par exemple, les résultats de la détection sont évalués relativement à une vérité terrain. Selon l'application envisagée, il peut être intéressant de délimiter le plus correctement possible le visage. Or la phase de reconnaissance peut s'appuyer sur la comparaison de zones spatiales très restreintes, par exemple les yeux. Ce cas de figure est illustré dans la figure 3.3.

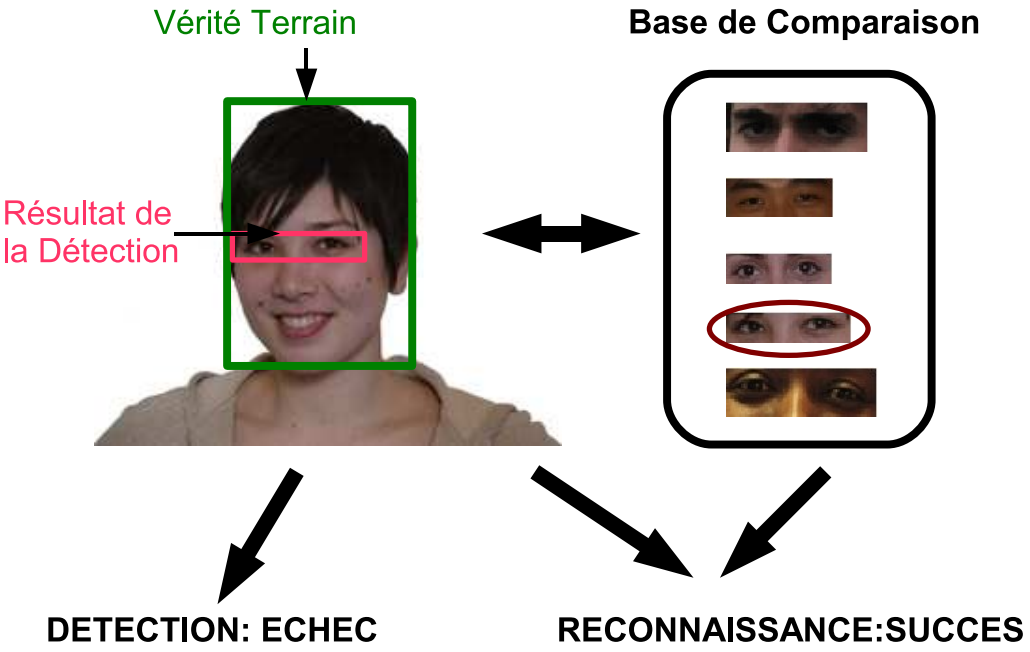


FIG. 3.3 – La réussite *globale* du système (au niveau de la reconnaissance) n'assure pas la réussite de chacun des modules (ici la détection)

En conséquence, nous préférons appliquer l'adaptation séquentiellement à chacun des modules (détection, suivi, amélioration et reconnaissance), mettant ainsi en jeu pour chacun de ces modules des critères d'évaluation propres à chacun d'entre eux. L'idée est ainsi de produire, dans

l'ordre de mobilisation dans la séquence du système, les paramètres optimaux de chaque module (notés respectivement $\tilde{\mathcal{E}}_D$, $\tilde{\mathcal{E}}_S$, $\tilde{\mathcal{E}}_A$ et $\tilde{\mathcal{E}}_R$ pour chacun de ces modules) en prenant à chaque fois en compte le paramétrage optimal produit pour le module précédent dans la séquence, tel qu'illustré dans la figure 3.4.

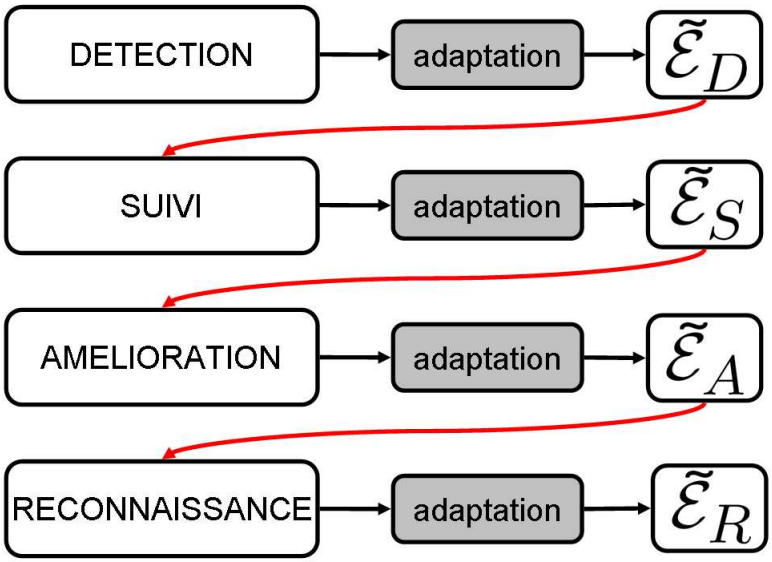


FIG. 3.4 – L'adaptation globale du système est conçue autour de l'adaptation séquentielle des 4 tâches qui le composent.

Il eût été possible d'appliquer cette méthodologie d'adaptation séquentielle à un niveau plus profond dans la hiérarchie. Pour autant, la nécessité de définir des mesures d'évaluation cohérentes et compréhensibles des résultats produits nous limite au niveau de représentation choisi. En effet, à d'autres niveaux, la construction des vérités terrains (les résultats attendus des différents modules), utilisées pour produire les mesures d'évaluation, poserait de nombreux problèmes (comment définir par exemple, la vérité terrain des résultats attendus d'un détecteur de contours ?).

Une dernière contrainte relève alors du mode de décomposition de chacun des quatres modules considérés. Nous imposons alors de choisir pour chacun d'entre eux une décomposition telle que chacun des modules la constituant produisent des résultats de même nature que les résultats finaux du module (une image dans laquelle sont isolées les zones contenant un objet ou un ensemble de boîtes englobantes pour la détection, des zones spatio-temporelles pour le suivi, une liste d'images ou une unique image pour l'amélioration et enfin un identifiant pour la reconnaissance).
Nous verrons en effet par la suite que l'analyse du fonctionnement des différents modules lors de l'établissement du diagnostic de responsabilité impose cette contrainte.

3.3.4 Contraintes sur la nature des objets

Tous les objets ne sont pas pris en compte dans cette étude. La méthodologie proposée se limitera aux systèmes DSRO dont l'objet d'étude est (ou peut être) :

1. connexe

2. non déformable
3. sémantiquement focalisé (un objet ne peut être "une peinture")
4. qualifié par un identifiant unique
5. **extractible** : le système étudié doit produire des résultats d'une qualité minimale pour que l'optimisation seule puisse améliorer les résultats. Dans le cas contraire, l'optimisation telle qu'elle est appréhendée ici ne suffit pas : une refonte totale du système, objectif s'établissant au delà des limites de notre méthodologie, devient nécessaire.

A titre d'exemple, les systèmes d'extraction de nuages [CSFN02, PBR98], ou d'ombres [PCMT01], qui sont des objets déformables, ne seront pas pris en compte. Cette contrainte s'associe par ailleurs évidemment avec l'obligation de choisir un corpus dans lequel les objets sont extractibles par le système : adapter un système d'extraction de visages de face sur un corpus composé uniquement de visages de profil relève d'une gageure.

3.3.5 Contraintes sur le corpus d'adaptation

L'objectif est d'ajuster les paramètres d'un système pour optimiser ses performances sur un corpus d'application CPS_{appli}. En vue de réduire le coût de l'adaptation, l'idée est alors de tirer profit du mode d'organisation des bases de documents à l'INA en termes de collections homogènes. Ainsi, on considérera par la suite que ces collections présentent une homogénéité suffisante du point de vue des objets qu'elles contiennent. En conséquence, pour un corpus d'application correspondant à une telle collection, l'adaptation pourra être limitée à un corpus d'adaptation CPS_{adapt} de taille réduite, considérant que l'homogénéité de CPS_{appli} permet d'étendre les résultats obtenus sur CPS_{adapt} à l'ensemble du corpus CPS_{appli}.

3.4 Organisation de la méthodologie

Il a déjà été expliqué que la méthodologie d'adaptation s'organisait autour de deux analyses distinctes : l'"*analyse des comportements*" et le "*diagnostic de responsabilité*". L'organisation autour de ces deux analyses constitue un premier apport de la méthodologie. Développons maintenant le fonctionnement de ces deux analyses dont les détails seront donnés dans la partie suivante, au cours de deux chapitres distincts qui auront autant pour objectif de justifier le fonctionnement choisi que les choix technologiques effectués. Etant donné que la méthodologie est appliquée séquentiellement aux quatres modules précédemment cités, l'adaptation désignera par la suite celle d'un module et non du système en son ensemble.

3.4.1 L'analyse des comportements

Cette première analyse repose sur le postulat selon lequel il existe, pour un unique corpus d'adaptation CPS_{adapt}, non pas un seul mais plusieurs ensembles de paramétrages optimaux du module considéré. Ces différents paramétrages sont associés à différents comportements, c'est à dire à différents types d'erreurs produits par le système. L'enjeu de l'analyse des comportements est alors de parvenir à distinguer les différentes catégories d'erreurs.
Pour ce faire, l'idée est de définir pour chaque module un ensemble de mesures d'évaluation qui permettront, en comparant les résultats produits par le module avec les résultats attendus (les vérités terrain), estimés sur CPS_{adapt}, de produire des **vecteurs de performances**. Une étape de **clustering**, relativement à ces vecteurs permettra finalement d'isoler les catégories homogènes

d'erreurs.
L'analyse des comportements nécessite donc de définir :

1. un ensemble de mesures d'évaluation relatives à chaque module,
2. des vérités terrains,
3. une méthode de clustering adéquate,

3.4.2 Le diagnotic de responsabilité

Selon le mode de représentation des systèmes adopté, chaque module (détection, suivi, amélioration et reconnaissance) peut être représenté comme une séquence d'autres modules de niveau inférieur. L'enjeu du diagnostic de responsabilité est alors de déterminer, pour chaque comportement extrait lors de l'analyse précédente, le module (de niveau inférieur) **responsable** de l'erreur constatée. L'idée est alors de définir un ensemble de caractéristiques visuelles qui permettront de représenter les objets sur lesquels le système produit chaque classe de comportement. Par la suite, les performances des différents modules de niveau inférieur sont étudiées relativement à chaque caractéristique choisie. La mise en relation des représentations des objets composant les différents comportements avec cette analyse des performances des modules de niveau inférieur, permettra par la suite de calculer pour chaque module de niveau inférieur un **indice de responsabilité** relativement à chaque caractéristique. Une méthode d'intégration des différents indices obtenus permettra finalement de déterminer pour chaque comportement, la nature du module de niveau inférieur responsable.
En conclusion, il sera donc nécessaire de définir pour effectuer ce *diagnostic de responsabilité* :

1. un ensemble de caractéristiques pour représenter les comportements,
2. pour chaque caractéristique, une base d'objets dont la variation de la caractéristique considérée est maîtrisée en vue de pratiquer l'analyse des performances des modules de niveau inférieur,
3. un indice de responsabilité intégrant la nouvelle représentation des comportements et l'analyse des performances des modules de niveau inférieur,
4. une méthode d'intégration des indices produits pour déterminer le diagnostic final de responsabilité.

3.4.3 Schéma global de fonctionnement de la méthodologie

Les figures 3.5 et 3.6 résument alors l'organisation générale de la méthodologie, détaillée dans les deux chapitres de la partie suivante. Dans ces figures, F^{visu} désigne l'ensemble des caractéristiques visuelles choisies pour représenter les objets composant les comportements, et M_{resp} désigne les modules responsables déterminés pour les différentes classes de comportements.

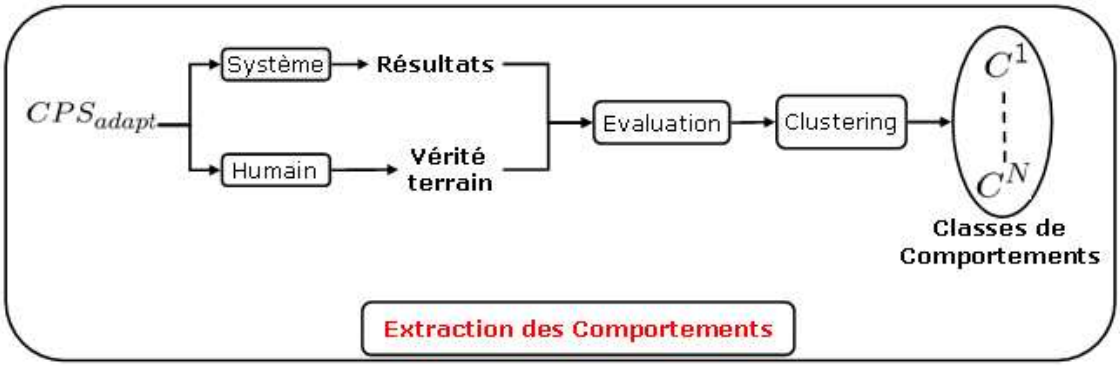


FIG. 3.5 – L'analyse (ou extraction) des comportements

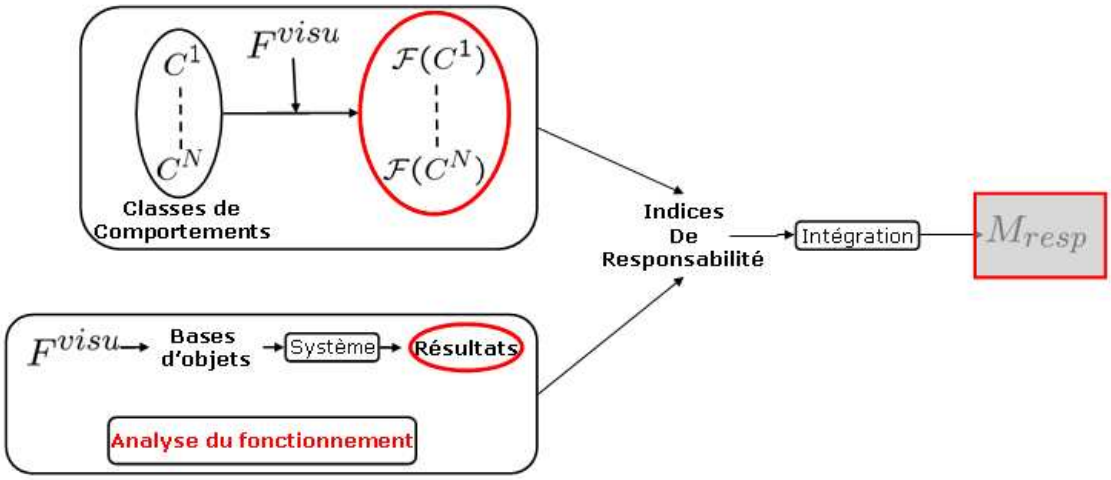


FIG. 3.6 – Le diagnostic de responsabilité

Deuxième partie

Méthodologie

1

Analyse des comportements d'un système DSRO

Sommaire

L'*analyse des comportements* repose sur le postulat selon lequel il existe autant de paramétrages optimaux d'un module (détection, suivi, amélioration ou reconnaissance) d'un système DSRO que celui-ci produit, sur le corpus d'adaptation CPS_{adapt}, de classes de comportement différentes, où ces classes sont définies selon la définition 9.

Définition 9. *Une classe de comportement correspond à un ensemble d'objets sur lesquels un module produit des performances équivalentes.*

L'extraction de ces classes impose dans un premier temps, de définir une métrique d'évaluation des résultats du module considéré. Pouvant être empiriques, les métriques proposées (une pour chaque module composant le système DSRO) reposent au contraire sur la comparaison des résultats avec une vérité terrain (cf définition 10).

Définition 10. *La vérité terrain d'un module, relativement à un document, correspond aux résultats attendus (les résultats idéaux) du module sur ce document. Pour la détection par exemple, la vérité terrain correspond à l'ensemble des zones englobantes des objets contenus dans les images du document (ou l'ensemble des images dans lesquelles sont isolés les objets) .*

L'application de ces différentes mesures permet alors d'associer à chaque objet de la vérité terrain un vecteur de performance. Un algorithme de clustering, s'appuyant sur cette représentation des objets permet finalement de constituer les classes de comportement.
La plan de ce chapitre s'impose alors de lui-même : après avoir développé plus formellement les motivations de cette analyse, seront abordées les questions de la définition de métriques d'évaluation, de la construction des vérités terrain, et enfin du choix d'un algorithme de clustering *ad hoc*. Bien que quelques unes des réflexions apportées dans ce chapitre puissent être étendues au cas général des systèmes de vision, la portée de cette étude sera le plus souvent limitée aux seuls systèmes DSRO. Par ailleurs, la problématique de l'analyse des comportements sera ici illustrée en se basant sur les résultats d'un système de détection de visages.

1.1 Motivations

Il est relativement courant que l'évaluation des systèmes de vision se limite à établir une distinction entre les résultats *corrects* et *incorrects*. Dans le cas des systèmes DSRO, chacun des objets de la vérité terrain peut être ainsi qualifié au regard des performances des différents modules : "correctement détecté", "correctement suivi", etc. Pour chacun des modules, la détermination des objets sur lesquels les résultats obtenus sont *corrects* s'appuie généralement sur le seuillage d'une mesure d'évaluation ; le couple formé par la mesure adoptée et le seuil qui lui est associé correspond alors à l'instanciation des critères d'évaluation établis lors de la formulation du problème auquel est censé répondre le système.
Cette solution s'avère insuffisante dans la perspective de l'optimisation. En effet, lorsqu'une unique classe correspondant aux échecs est constituée, il n'existe pas nécessairement de cohérence en ce qui concerne la manifestation de l'erreur mesurée. Ceci va alors à l'encontre du postulat suivant.

Postulat 1. *La nature des paramètres à modifier pour optimiser un système dépend d'une qualification précise de l'erreur constatée.*

Ce postulat a déjà pu être illustré dans les images de la figure 1.5 dans la partie 1. Il avait déjà été souligné alors que la manifestation de l'erreur constatée (sur les images a et c) n'était pas la même : d'un côté "trop de fausses alarmes" et de l'autre "trop d'oublis". Il a été par ailleurs montré qu'il était possible d'améliorer les résultats sur chacune de ces deux images (cf figure 1.8) en mettant en oeuvre sur celles-ci deux paramétrages différents du système. Circonscrire les échecs à une unique classe de comportement aboutit à la production d'un paramétrage "*optimal*" unique dont il est évident qu'il aboutira à des résultats non satisfaisants, puisque deux paramétrages différents au moins sont nécessaires. Cette idée se trouve d'ailleurs justifiée par les images de la figure 1.1, montrant les résultats obtenus sur les images a et c de la figure 1.5, en appliquant le paramétrage optimal trouvé pour l'autre image.

(a) Résultat obtenu en appliquant sur l'image a le paramétrage optimal de l'image c

(b) Résultat obtenu en appliquant sur l'image c le paramétrage optimal de l'image a

FIG. 1.1 – Une illustration de l'existence de différents paramétrages optimaux

Ces résultats montrent ainsi que la solution idéale consiste à produire plusieurs ensembles de paramètres optimaux, pour chacune des classes formées relativement à la *forme* des erreurs constatées.

La nécessité de différencier plusieurs classes de résultats s'illustre de nouveau si le système entier de détection de visages proposé dans [SP96] est mis en oeuvre. A la suite du filtre HSL déjà évoqué, les composantes connexes de l'image sont extraites et filtrées selon des critères géométriques. La détermination des ellipses englobantes de ces composantes permet enfin de délimiter les visages, sachant que les zones proposées sont une nouvelle fois filtrées selon leur taux de recouvrement avec les composantes auxquelles elles sont associées. Une unique modification a été apportée au système afin d'améliorer le filtrage final des zones : pour faciliter le choix du seuil utilisé (établi entre 0 et 1, à la différence de la méthode initiale qui ne permettait pas de s'assurer des bornes de ce dernier). Le critère de sélection manipulé est ainsi modifié selon l'équation 1.1 :

$$\begin{aligned} 1.\ & \frac{A(C \cap E)}{A(E)} > TH \\ 2.\ & \frac{A(C \cap E)}{A(C)} > TH \end{aligned} \tag{1.1}$$

où C et E désignent respectivement les composantes et ellipses associées et A l'aire de ces zones. Le système est appliqué sur 100 images issues d'un journal télévisé pour différents seuils TH. Le seuil de 0.4 est alors retenu en vue de limiter l'élimination de zones de visages correctement détectées. Les figures 1.2, 1.3 montrent respectivement la représentation hiérarchique du module de détection de visage utilisé ainsi qu'un exemple de résultat.

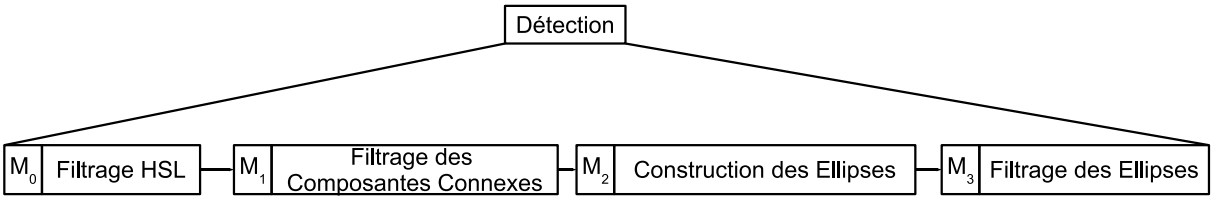
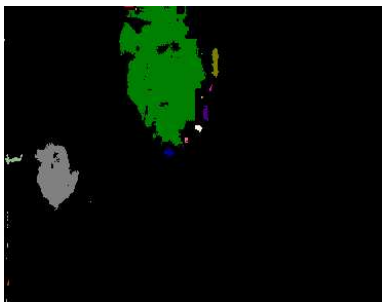


FIG. 1.2 – Les différents modules du système de détection de visages utilisé

(a) Image Originale



(b) Les différentes composantes connexes extraites après le filtre HSL

(c) Les ellipses englobantes des composantes retenues après le filtrage géométrique

FIG. 1.3 – Résultats des différentes étapes du détecteur de visage

Une vérité terrain des visages est constituée sous la forme d'ellipses englobantes de ces derniers. La mesure d'évaluation utilisée repose alors sur les taux de recouvrement entre les zones détectées et les zones de la vérité terrain comme détaillé dans la formule 1.1. Les images des figures

1.4, 1.5, 1.6, 1.7, 1.8 et 1.9 illustrent les résultats obtenus pour un seuil de 0.5 [10]. L'existence de différents comportements parmi les résulats "incorrects" est ici démontrée visuellement. On distinguera ainsi parmi les comportements, les oublis, les visages pour lesquelles la zone produite par le système est trop haute ou trop petite, etc.

(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat

(e) Vérité terrain (f) Résultat

FIG. 1.4 – Des exemples de résultats corrects de détection

(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat

FIG. 1.5 – Le cas des oublis

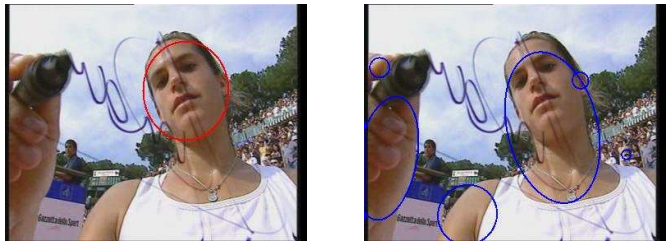
A l'issue de ces expérimentations, l'existence de différents comportements est donc prouvée et doit être prise en compte lors de l'adaptation. La question de l'obtention des différentes classes de comportements est alors soulevée. La méthode la plus immédiate met en jeu la subjectivité du concepteur lorsque ce dernier prend le parti de déterminer *a priori* un lien entre la nature des objets à extraire et les résultats du système. Les classes sont alors constituées des objets

[10] Seuls les visages sur lesquels le résultat du module de détection est évalué sont délimités dans les images de la vérité terrain

(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat

FIG. 1.6 – Cas de segmentation

(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat



(e) Vérité terrain (f) Résultat

FIG. 1.7 – La zone détectée est trop haute

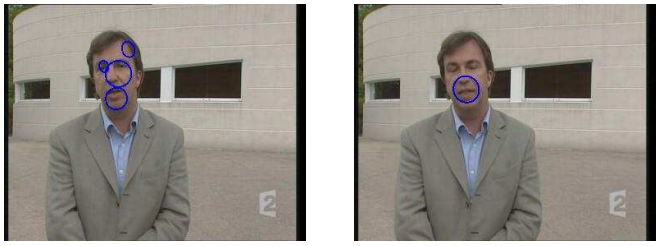
(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat

FIG. 1.8 – La zone détectée est trop petite (trop faible recouvrement)

(a) Vérité terrain (b) Résultat (c) Vérité terrain (d) Résultat

FIG. 1.9 – Deux comportements "*autres*"

similaires selon les critères retenus par le concepteur. Cette méthode suppose malheureusement de connaître une modélisation précise du fonctionnement du système relativement aux objets qui lui sont présentés en entrée, modélisation extrêmement difficile à concevoir. Etablir *a priori* la similarité entre deux objets en vue de constituer les classes se révèle une tâche par trop ambitieuse. Les images de la figure 1.10 donnent ainsi une illustration des risques encourus en assimilant les classes d'objets avec les classes de comportements : si le visage semble identique dans ces deux images, certaines différences très limitées, voire imperceptibles aboutissent à des résultats différents du système.



(a) (b)

FIG. 1.10 – Les résultats du système peuvent être différents sur deux images successives du flux : l'équivalence entre les classes d'objets et les classes de comportements n'est en aucune façon vérifiée.

Enfin, cette méthode de constitution des classes de comportement suppose que la seule cause potentielle d'échec repose sur le modèle des objets adoptés, alors que l'instanciation du filtre associé au modèle (c'est à dire le choix des opérateurs constituant le système) peut aussi être mise en cause.
La détermination des classes de comportements se doit donc de reposer sur les données issues de l'évaluation du système, c'est à dire de la comparaison de ses résultats avec une vérité terrain. La définition d'une mesure d'évaluation adaptée aux différents modules des DSRO, la construction des vérités terrains ainsi que le choix d'un algorithme de clustering constituent alors les points centraux de l'analyse des comportements et seront présentés dans les parties suivantes.

1.2 La question de la mesure d'évaluation : considérations générales et état de l'art

1.2.1 Quelques points généraux

1. **Le problème de la non prédictibilité des systèmes de vision**
   La principale difficulté à évaluer un système de vision provient de ce que les résultats de ce dernier sont non prédictibles. En effet, il a déjà été expliqué qu'il était difficile voire impossible d'établir une représentation analytique de tels systèmes. Cette non prédictibilité a pour conséquence de restreindre la portée des conclusions de toute évaluation aux seules images (ou vidéos) sur lesquelles le système a été appliqué ou à une classe d'images (ou de vidéos) similaires selon la *distance orientée objet*. La tâche d'évaluation est donc locale (au sens de l'espace des données d'entrée d'un système).
   Par ailleurs, la non prédictibilité ajoute à la difficulté de concevoir une mesure [11] d'évaluation robuste en ce sens qu'il n'est pas suffisant de se limiter à l'observation des comportements du système sur un nombre limité d'entrées pour en déterminer les fondements. Dans le cas de la détection par exemple, il peut être nécessaire de prendre en compte, pour définir la métrique d'évaluation, les cas de fusion et de segmentation qui seront précisés par la suite, cas qui peuvent ne pas être constatés sur un ensemble limité d'observations.
2. **L'évaluation et le contexte applicatif**
   Définir une mesure d'évaluation empirique (qui ne s'appuie pas sur la comparaison avec une vérité terrain) est une tâche particulièrement complexe qui ne sera pas abordée dans cette étude. La définition d'une mesure d'évaluation se ramène donc à la construction d'une distance[12] entre les résultats et la vérité terrain.
   On peut alors distinguer deux types de systèmes : ceux pour lesquels la construction d'une vérité terrain est possible (c'est le cas des systèmes DSRO), des autres systèmes pour lesquels la tâche s'avère soit trop fastidieuse, soit tout à fait impossible (système d'amélioration de la qualité visuelle d'une image ou de détection de contours tant la notion de contour est mal définie). Dans tous les cas, et surtout dans le second, il est utile de se rapporter au contexte applicatif du système, c'est à dire à l'emploi envisagé de ses résultats pour parvenir à définir une mesure d'évaluation cohérente. Par exemple, dans le cas de la détection de contours, les contours extraits peuvent être la base d'un système plus vaste de représentation plus condensée des images. L'évaluation repose alors sur des critères psychovisuels. Les mêmes critères peuvent d'ailleurs être utilisés dans le cadre de l'évaluation d'un système d'amélioration de la qualité visuelle où ils seront par ailleurs fonction du scénario d'utilisation des résultats envisagé (support (télévision, cinéma, etc.), public visé). Dans le cas des systèmes DSRO, les applications envisagées peuvent être relatives à la documentation : segmentation, etc. Ces applications peuvent aussi être beaucoup plus précises : délimiter tous les visages des présentateurs, reconnaître tous les noms propres, ...

[11] La dénomination mathématique de mesure est souvent utilisée à tort. Strictement une mesure se définit de la façon suivante :

Définition 11. *Etant donné $(\mathcal{X}, \mathcal{A})$ mesurable (i.e : $\mathcal{A}$ est une tribu sur X), on appelle mesure une application $\mu : \mathcal{A} \to [0, +\infty]$ telle que :*

- $\mu(\emptyset) = 0$
- *Si les A_i sont disjoints, au plus dénombrables, alors $\mu(\bigcup A_i) = \sum \mu(A_i)$*

Dans le cas où ces axiomes ne pourraient être vérifiés, on préfèra donc la terminologie de métrique ou d'indice.

[12] Même remarque que pour la terminologie de mesure

Dans ce second cas, les contraintes à appliquer à la mesure d'évaluation sont très strictes : seuls les objets recherchés doivent être pris en compte (ici les visages des présentateurs ou les noms propres)[13].

3. **L'évaluation et le contexte visuel**
   La mesure d'évaluation peut, ou non, prendre en compte le contexte visuel, c'est à dire pondérer sa sortie en fonction de la difficulté de l'objet considéré à être détecté, suivi, etc..
4. **Composition d'une mesure d'évaluation**
   Il est courant d'établir la mesure d'évaluation par la fusion de différents indices. La mesure est alors plus ou moins précise selon leur nombre et la façon dont est appliquée la fusion. Généralement cette dernière repose sur le calcul de la norme d'un vecteur incluant ces indices. Le cas de la composition est illustré dans la figure 1.11.

FIG. 1.11 – Deux résultats de détection (fictifs) sur une même image : si la mesure d'évaluation prend uniquemenent en compte la somme des aires des zones recouvrant la vérité terrain, les deux résultats sont comparables. L'ajout d'une composante "segmentation" à la mesure permet de les différencier

1.2.2 Etat de l'art des métriques d'évaluation des systèmes DSRO

L'adaptation proposée est séquentielle. L'évaluation porte donc sur chacun des modules séparément et l'état de l'art proposé ici se construit de la même façon.

1.2.2.1 Evaluation du module de détection

Il convient de comparer pour chaque image, l'ensemble des zones détectées par le système avec celui conservé dans la vérité terrain.
Il arrive alors parfois que les modalités d'évaluation des systèmes ne soient pas décrites (en détection de visages par exemple, il est reconnu que le critère de réussite de la phase de détection est assez fréquemment passé sous silence dans les articles [YKA02]). Les résultats peuvent aussi être validés par une simple inspection visuelle comme par exemple dans le cadre de la détection

[13] Une mesure d'évaluation, guidée par des usages comparables, a été développée pour les systèmes commerciaux d'OCR (*Optical Character Recognition*). Quelque peu en marge des préoccupations de cette étude, ces travaux seront détaillés dans une annexe à ce manuscrit.

de constructions sur des images aériennes [Shu99]. Au delà de la subjectivité d'une telle évaluation, on soulignera la relative impossibilité d'appliquer un tel "protocole" à un corpus de test important.
En ce qui concerne l'évaluation quantitative, le principe le plus simple consiste à comparer les taux de recouvrement mutuels entre les zones détectées par le système et les zones conservées dans la vérité terrain comme dans l'équation 1.1, dont chaque membre est réécrit dans l'équation 1.2 en fonction des notations plus générales adoptées :

$$\frac{A(G_j \cap D_i)}{A(D_i)} \; et \; \frac{A(G_j \cap D_i)}{A(G_j)} \tag{1.2}$$

où D_i et G_j désignent respectivement une zone détectée et une zone de la vérité terrain. $A(Z)$ correspond à l'aire de la zone Z. En appliquant ensuite un seuil à ces deux critères, il est possible de déterminer les fausses alarmes (éléments détectés par le système associés à aucun objet de la vérité terrain) et les oublis (éléments de la vérité terrain associés à aucun objet détecté par le système). Les mesures de précision, rappel et de moyenne harmonique, issues de travaux en recherche de l'information [vR79] (cf 1.3) sont alors utilisées.

$$\begin{aligned} Précision &= \frac{Nombre\ d'objets\ correctement\ détectés}{Nombre\ total\ d'objets\ détectés} \\ Rappel &= \frac{Nombre\ d'objets\ correctement\ détectés}{Nombre\ d'objets\ à\ détecter} \\ MH &= \frac{2 \times Précision \times Rappel}{Précision + Rappel} \end{aligned} \tag{1.3}$$

Il existe plusieurs variantes à ce protocole. L'une d'entre elles, utilisée dans [BBTR04], repose sur la mesure suivante :

$$\frac{A(G_i \cap D_j)^2}{A(G_i)A(D_j)}$$

La comparaison des boîtes englobantes détectées par le système avec celles de la vérité terrain est la façon la plus courante d'établir une évaluation des résultats de la détection, que ce soit pour les textes [COB04, LW02], les voitures [PBC97] ou encore les objets en général [MMP+02]. Pour autant, la définition d'un seuil concernant les taux de recouvrement minimaux à partir duquel un objet est considéré comme correctement détecté est souvent subjective et il est courant que la question de son choix ne soit pas abordée.
Dans le cas des visages, la position des boîtes englobantes peut être déterminée à partir de celle des yeux par des contraintes anthropométriques. Certains systèmes recherchent donc explicitement cette partie du visage pour identifier une zone de l'image comme étant un visage. Un objet est alors repéré spatialement par deux zones distinctes. [PRTM04] propose une mesure d'évaluation dans ce cas de figure. La *distance* entre la vérité terrain et les résultats du système est alors évaluée par un ensemble de quatre valeurs (3 rapports de distance et un angle). Chacune des ces composantes est évaluée séparément et finalement les 4 mesures obtenues sont fusionnées par une combinaison linéaire dont les coefficients correspondent aux poids attribués à chacun des critères. La mesure d'évaluation utilisée pour chaque composante est de la forme $\Psi(x, \Theta)$, où Θ désigne un ensemble de trois paramètres définissant la rigueur de l'évaluation. La mesure globale obtenue est finalement seuillée pour déterminer les visages correctement détectés. Enfin, il convient de noter que la rigueur de l'évaluation est fonction des objectifs du système : détection ou localisation ; la seconde opération étant plus précise que la première.

L'étude proposée dans [PRTM04] préfigure les cas d'*associations multiples* (*multiple-matching*) traités dans plusieurs travaux [HWZ04a, MMP+02, Wol03]. La mesure décrite dans [Wol03] s'appuie sur la proposition exposée dans [LPH97]. Deux matrices, σ *et* τ, sont définies de la façon suivante :

$$\sigma_{ij} = \frac{A(G_i \cup D_j)}{A(G_i)} \; et \; \tau_{ij} = \frac{A(G_i \cup D_j)}{A(D_j)}$$

Les mesures de précision et de rappel sont alors calculées à partir de ces matrices en assignant des coûts en fonction du cas observé : une (plusieurs) zone(s) de la vérité terrain est(sont) associé(es) à une(plusieurs) zone(s) détecté(es) par le système. Dans [MMP+02], sept mesures différentes (au niveau des pixels ou des aires des zones englobantes), toutes basées sur les critères de l'équation 1.2, sont proposées. Les cas de fragmentation sont pris en compte et un indice spécifique est défini :

$$Frag(G_i^{(t)}) = \begin{cases} indéfini \; si \; N_{D^{(t)} \cap G_i^{(t)}} = 0 \\ \frac{1}{1+log_{10}(N_{D^{(t)} \cap G_i^{(t)}})} \end{cases}$$

Les mesures proposées au niveau des images peuvent, par ailleurs, être étendues à l'ensemble des images du flux sur lequel le système est appliqué.
Dans [HWZ04a] enfin, la mesure d'évaluation est très complexe et nécessite de construire des vérités terrain comportant de nombreuses informations (des indices de reconnaissabilité sont par exemple nécessaires, la longueur du texte ainsi que la variation de la hauteur des caractères doivent aussi être calculés). En ce qui concerne l'aspect *multiple-matching*, seule la fragmentation est prise compte par le calcul d'un indice de qualité de la fragmentation.

1.2.2.2 Evaluation du module de suivi

L'évaluation du module de Suivi repose sur des critères spatiaux-temporels. La vérité terrain est spatio-temporelle et correspond à l'ensemble des instances d'un même objet (on parlera de séquence d'un objet). Les erreurs peuvent être de deux types : fragmentation (spatiale ou temporelle) ou fusion (spatiale ou temporelle). La construction d'une mesure efficace implique donc de considérer le cas le plus complexe d'association entre les résultats du système et les données conservées dans la vérité terrain, celui dit de "*many-to-many*" dans la terminologie anglosaxone (ou "*relation M-M*" en français, avec, implicitement, M>1). Ce cas est abordé dans [BST+05] où le processus d'évaluation se déroule en deux phases distinctes :

1. Pour chaque séquence détectée par le système, la liste des séquences de la vérité terrain avec lesquelles il existe une intersection spatio-temporelle est établie. Les taux de recouvrement spatio-temporels obtenus sont ensuite cumulés. L'examen de ceux-ci permet de déterminer les cas de fausses-alarmes et les erreurs de fusion.
2. Pour chaque séquence de la vérité terrain, la liste des séquences établies par le système avec lesquelles il existe une intersection spatio-temporelle est établie. De la même façon que dans la première phase, si l'intersection spatio-temporelle cumulée est inférieure à un certain seuil, la séquence de la vérité terrain considérée est un oubli. Dans le cas contraire, on est dans le cas d'une erreur de fragmentation.

La construction des vérités terrains est souvent considérée comme un frein à une évaluation extensive des systèmes. Pour s'affranchir des limitations inhérentes à la manipulation de vérités terrain, [WZ04] présente une méthode d'évaluation autonome originale, basée sur plusieurs mesures de cohérence des séquences détectées par le système (ce dernier a pour objectif le suivi de véhicules et de personnes). Les critères de cohérence se rapportent à :

1. la complexité de la trajectoire,
2. la souplesse du mouvement,
3. la cohérence de l'échelle,
4. la similarité de la forme,
5. la cohérence de l'apparance (i.e : au niveau des pixels) entre les différents éléments de la séquence

Un critère global d'évaluation est ensuite construit par pondération de ces critères de cohérence. Il est à remarquer que ces différentes mesures font intervenir un ensemble de 11 seuils dont la détermination est difficile ce qui remet en cause la validité de la méthode. L'auteur propose en effet pour fixer ces seuils d'utiliser une méthode par apprentissage ce qui implique *a posteriori* la construction de vérités terrain. Une autre critique porte sur le choix des critères de cohérence, reposant uniquement sur des *a priori* : "*normalement, un véhicule ne change pas de vitesse et de direction de façon très marquée entre deux images*".
L'évaluation peut aussi être centrée sur le système et s'appuyer sur les associations entre les suiveurs ("*trackers*") associés aux objets et ces derniers. Ainsi, si un objet est associé à différents suiveurs, la tâche de suivi a échoué. Ce mode d'évaluation est proposé dans [SGPOB05] (cf figure 1.12).

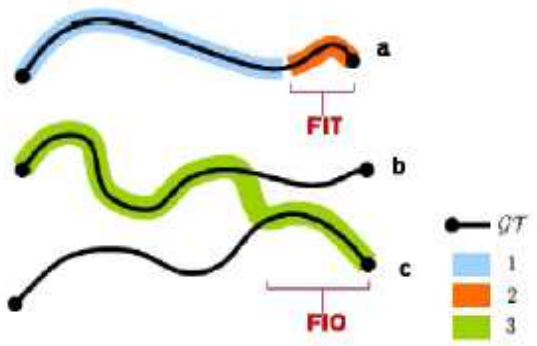


FIG. 1.12 – Deux erreurs de suivi différentes d'après [SGPOB05] (*FIT* et *FIO* désignent les types d'erreurs rencontrés et *GT* correspond à la réduction de *Ground Truth* (vérité terrain)).

Les acronymes *FIT* et *FIO* désignent les deux erreurs de suivi prises en compte, soit, en anglais : *Falsely Identified Tracker* et *Falsely Identified Object*. La première erreur intervient sur l'objet a, suivi par deux trackers différents. Le second type d'erreur est illustré avec le suivi des objets b et c : le même tracker change de cible pendant le suivi de l'objet b pour passer au suivi de l'objet c. Ces cas se rencontrent lorsque deux objets se croisent et que le tracker lié à un objet prend alors un point d'accroche sur l'objet croisé (un tracker est assigné à l'objet qu'il suit le plus longtemps). Selon des critères concernant les taux de recouvrement temporels entre le tracker et les objets, ces deux types d'erreurs sont pris en compte pour établir une mesure globale d'évaluation.

1.2.2.3 Evaluation du module d'Amélioration

Les résultats de l'amélioration doivent être évalués en termes d'amélioration des résultats de la reconnaissance. Une méthode d'amélioration de la qualité des images d'empreintes est proposée dans [HWJ98]. L'évaluation repose alors sur la comparaison entre les résultats du système

sans et avec la phase d'amélioration. Deux tâches sont évaluées : l'extraction des *minuties* (*minutiae*) qui correspondent à des points caractéristiques sur les empreintes, ainsi que la vérification d'empreinte. Dans le premier cas, la mesure d'évaluation prend en compte un indice précisant le degré de difficulté de la tâche, relativement à la qualité de l'image sur laquelle celle-ci est appliquée. Dans le cas de la vérification, la mesure d'évaluation repose sur la mesure des fausses alarmes et des vérifications réussies.
Une étude de l'incidence de l'amélioration de la qualité des images est proposée de la même façon dans [Sum01] où les résultats de différents OCR [14] sont comparés en fonction de la qualité des textes qui leur sont donnés en entrée (bruit, résolution, ...).

1.2.2.4 Evaluation du module de Reconnaissance

La nature des résultats de la reconnaissance est fonction de la nature de l'objet considéré. Dans le cas du texte, la reconnaissance produit une chaîne de caractères et l'évaluation consiste alors à comparer celle-ci avec la chaîne conservée dans la vérité terrain. Dans le cas d'objets dont l'identification textuelle relève d'une interprétation de très haut niveau (visages, voitures, etc), la phase de reconnaissance s'appuie sur la comparaison entre la zone détectée et une base de données contenant un ensemble d'objets pré-identifiés, l'objet détecté étant associé à l'objet de la base avec lequel la distance obtenue est minimale. Nous proposons ici de revenir sur la problématique de l'évaluation du module de reconnaissance relativement à deux objets pour lesquels cette tâche s'avère d'importance relativement au cadre applicatif de l'aide à la documentation : les visages et les textes.
Le protocole d'évaluation FERET [RPM02, PMRR00] distingue l'évaluation des systèmes d'identification (un visage inconnu est présenté à un algorithme qui doit le reconnaître) de celle des systèmes de vérification (un visage est présenté à un algorithme avec une identité et celui-ci doit déterminer si cette identité est juste). Le protocole d'évaluation met l'accent sur la nature des données sur lesquelles les systèmes sont appliqués : différentes combinaisons concernant la nature des prises de vue sont prises en compte ainsi que différentes constructions des ensembles considérés, à savoir l'ensemble cible et l'ensemble de requête. Dans le cas de l'identification (plus proche de nos considérations), l'évaluation repose sur une mesure très simple basée sur la comparaison des scores d'identification obtenus par le système en comparant l'image requête avec toutes les images proposées dans l'ensemble cible.
L'évaluation de la reconnaissance est abordée selon le même axe dans [LBF05] : à partir de deux ensembles (*gallery set* et *probe set*) les sytèmes de reconnaissance d'iris sont évalués successivement pour établir l'influence de la nature de ces ensembles. La qualité des images utilisées est aussi prise en compte. La mesure d'évaluation se limite aux taux de reconnaissance au rang 1 (pour une requête image, le rang 1 correspond au taux d'identification le plus élevé).
En ce qui concerne le module de reconnaissance d'un système DSRO dédié au texte, l'évaluation repose sur la comparaison de la chaîne s_{res} produite par le système avec celle conservée dans la vérité terrain, s_{vt}. Dans ce cas, la mesure d'évaluation la plus couramment utilisée est la mesure de Levenstein (ou distance d'édition) [WF74]. Le principe consiste à calculer le coût minimal ($\delta(s_{vt}, s_{res})$) en termes de transformations nécessaires pour convertir la chaîne s_{res} en la chaîne de la vérité terrain s_{vt} . Les transformations prises en compte sont la substitution, l'insertion et

[14] Optical Character Recognition

la suppression d'un caractère, auxquelles sont associés les coûts suivants :

$$\begin{aligned}
&Substitution\ (a \rightarrow b) : coût\ \gamma(a,b)\\
&Insertion\ (\lambda \rightarrow b) :\ coût\ \gamma(\lambda,b)\\
&Suppression\ (a \rightarrow \lambda) :\ coût\ \gamma(a,\lambda)\\
&avec\ \delta(a,b) = \gamma(a,b)
\end{aligned}$$

La détermination des coûts est alors à l'appréciation de l'utilisateur. Pour comparer les résultats entre des chaînes de longueurs différentes, $\delta(s_{vt}, s_{res})$ doit être normalisé en fonction du nombre de caractères contenus dans s_{vt}. Il est à remarquer que des mesures de précision et de rappel peuvent ensuite être définies en fonction du nombre de caractères correctement reconnus [LW02, WMR99] et qu'il est possible de distinguer le taux de reconnaissance des mots de celui des caractères [WMR99].

1.3 Définition des mesures d'évaluation adoptées

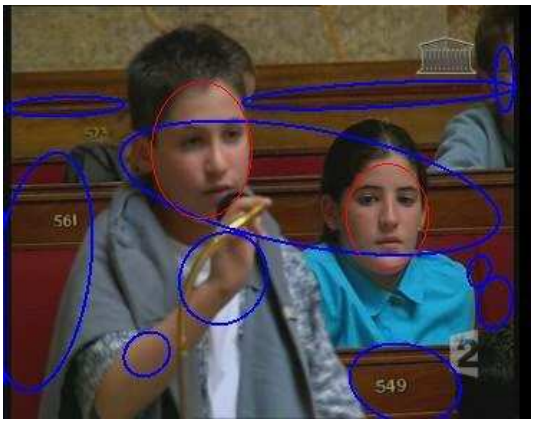
Chaque module d'un système DSRO se voit associé une mesure propre à son fonctionnement. Chacune de ces mesures a pour objectif d'établir un indice de réussite du module considéré à effectuer la tâche à laquelle il est assigné, sur un objet particulier. L'ensemble des données issues de l'évaluation porte donc sur le niveau "*objet*". Tous les objets de la vérité terrain se voient ainsi dotés d'un vecteur de mesures dont les composantes seront définies ci-dessous en fonction du module considéré.

1.3.1 Pour la détection

L'évaluation peut être établie à différentes granularités : du niveau des pixels à celui des objets. Le choix de la granularité dépend des données disponibles en sortie du module de détection. Quel que soit le niveau choisi, la mesure d'évaluation se doit de permettre d'identifier les faux positifs (les fausses-alarmes), les faux négatifs (les oublis) et les vrais positifs (les objets correctement détectés). La détermination de ces différentes classes de résultats impose l'usage de seuils dont la détermination est très difficile et souvent sujette à subjectivité. La solution la plus simple consiste alors à appliquer plusieurs seuillages différents, échos de la rigueur avec laquelle l'évaluation est pratiquée.
Globalement, la mesure nécessite :

1. d'associer entre eux, les objets de la vérité terrain et ceux détectés par le système, l'extraction des faux positifs et des faux négatifs étant effectuée dans le même temps par le seuillage précité,
2. de proposer un vecteur de mesures de performances pour chaque objet de la vérité terrain qui n'a pas été "*oublié*" par le système, vecteur devant donner mesure des cas d'associations multiples déjà évoqués (cf figure 1.13).

La méthode d'association la plus simple repose sur les éléments de l'équation 1.2. Les faux positifs et négatifs sont isolés en fonction du seuil utilisé.
La construction des vecteurs de performance repose ensuite sur les calculs suivants. Etant donné une image *Im* du flux, on notera par la suite $\{VT_i\}_i$ l'ensemble des zones de la vérité terrain et $\{Res_k\}_k$ l'ensemble des zones détectées par le système dans cette image. La mesure d'évaluation de la détection de chaque zone VT_i comporte alors quatre indices différents : deux d'entre eux correspondent aux taux de recouvrement mutuels de l'équation 1.2, un autre est lié aux positions



(a) Un cas de fusion : deux zones de la vérité terrain sont associées à une même zone détectée par le système

(b) Un cas de segmentation : plusieurs zones détectées sont associées au même objet de la vérité terrain

FIG. 1.13 – Deux cas d'association différents : la fusion et la segmentation

respectives des zones associées et un dernier indice est lié à la fusion. Chaque zone VT_i est associée à un ensemble de zones détectées E_i^{Res} et chaque zone détectée Res_k est associée à un ensemble de zones de la vérité terrain E_k^{VT}. Les cas d'associations multiples interviennent alors dans les situations suivantes :

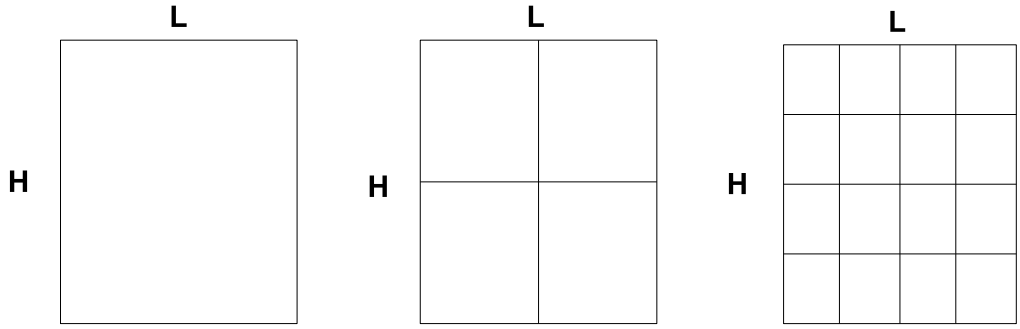
- Si $\#(E_i^{Res}) > 1$, il y a *segmentation*
- Si $\#(E_k^{VT}) > 1$, il y a *fusion.*

Quelle que soit la situation rencontrée, les trois premiers indices sont invariablement calculés selon le mode suivant :

1. $I^1_{recouv}(VT_i) = MOY_{Res_j \in E_i^{Res}}(I^1_{recouv}(VT_i, Res_j))$
   $où\ I^1_{recouv}(VT_i, Res_j) = \dfrac{A(VT_i \cap Res_j)}{A(VT_i)}$
2. $I^2_{recouv}(VT_i) = MOY_{Res_j \in E_i^{Res}}(I^2_{recouv}(VT_i, Res_j))$
   $où\ I^2_{recouv}(VT_i, Res_j) = \dfrac{A(VT_i \cap Res_j)}{A(Res_j)}$
3. $I_{pos}(VT_i) = MOY_{Res_j \in E_i^{Res}}(I_{pos}(VT_i, Res_j))$
   $où\ I_{pos}(VT_i, Res_j) = \|\overrightarrow{\mathcal{G}(VT_i)} - \overrightarrow{\mathcal{G}(Res_j)}\|_{L^2}$
   et $\mathcal{G}(VT)$ *désigne le centre de gravité d'une zone.*

On remarquera ici que $I^2_{recouv}(VT_i) = 1$ si $\forall j,\ Res_j \subset VT_i$.
Reste à savoir comment quantifier les évènements de segmentation et de fusion. En ce qui concerne la segmentation, sa prise en compte intervient dans I^1_{recouv} : plus le nombre d'éléments détectés entrant en compte est important, plus cet indice diminue comme illustré dans la figure 1.14.

Concernant la fusion, un nouvel indice I_{fusion} lui est consacré. Cet indice est toujours associé à un élément de la vérité terrain VT_I. Dans le cas de la fusion, une zone détectée Res_k qui lui est associée est aussi associée avec un ensemble d'autres zones de la vérité terrain. I_{fusion}



(a) $I^1_{recouv} = 1$ (b) $I^1_{recouv} = \frac{1}{4}$ (c) $I^1_{recouv} = \frac{1}{16}$

FIG. 1.14 – L'objet à détecter (le plus grand rectangle) est parfaitement recouvert par l'ensemble des rectangles produits par le système. Pour autant, l'indice I^1_{recouv} prend en compte le degré de segmentation et décroit en fonction du nombre de rectangle détectés entrant en jeu dans cette segmentation.

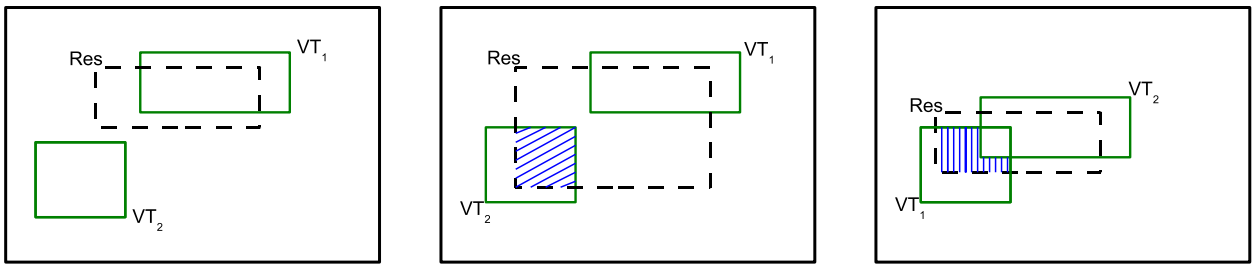
est alors calculé de la sorte :

$$I_{fusion}(VT_I) = \frac{\sum_{i \neq I} A(Res_k \cap VT_i)}{A(Res_k \backslash (Res_k \cap VT_I))}$$

Ce cas de figure correspond au cas idéal pour lequel, les zones de la vérité terrain ne s'intersectent pas. Dans le cas contraire, l'indice utilisé est plus restrictif et devient :

$$I_{fusion}(VT_I) = \frac{\sum_{i \neq I} A(Res_k \cap (VT_i \backslash VT_I))}{A(Res_k \backslash (Res_k \cap VT_I)}$$)

Ces différents cas sont illustrés dans la figure 1.15



(a) La zone détectée Res n'intersecte aucune autre zone de la vérité terrain que VT_1, $I_{fusion}(VT_1) = 0$

(b) Un cas simple : $VT_1 \cup VT_2 = \emptyset$, $I_{fusion}(VT_1)$ prend en compte l'aire du rectangle hachuré

(c) Un cas plus complexe : $VT_1 \cup VT_2 \neq \emptyset$, l'aire du rectangle entrant en jeu dans le calcul de $I_{fusion}(VT_2)$ correspond à $A(Res \cap (VT_1 \backslash VT_2)$

FIG. 1.15 – Les différents cas de figure de la fusion

Dans le cas le plus général, il existe pour une zone de la vérité terrain VT_I plusieurs zones Res_k sources de fusion. L'indice de fusion correspond alors à la moyenne des indices de fusion

obtenus pour les différentes zones Res_k. La mesure de détection adoptée d'un objet VT_i de la vérité terrain est donc finalement le vecteur à quatre composantes suivant :

$$\mathcal{M}^{detection}(VT_i) = Moy_{Res_j \in E_i^{Res}} \begin{pmatrix} I^1_{recouv}(VT_i, Res_j) \\ I^2_{recouv}(VT_i, Res_j) \\ I_{pos}(VT_i, Res_j) \\ I_{fusion}(VT_i, Res_j) \end{pmatrix} \tag{1.4}$$

La métrique adoptée ici suppose que chaque objet de la vérité terrain est délimité par une unique zone englobante. La question de l'extension de cette métrique au cas présenté dans [PRTM04] est alors soulevée. Le plus simple est de conserver cette même mesure, alors moyennée sur l'ensemble des zones incluses dans la délimitation des objets (typiquement les deux zones correspondant aux yeux dans le cas évoqué dans [PRTM04]). Par ailleurs il est alors nécessaire d'ajouter un autre indice à notre vecteur de performance, relatif au maintien de l'organisation des différentes zones délimitant l'objet. Cet indice doit rendre compte de la distortion de l'objet (représenté par les différentes zones de la vérité terrain) produite par le système. Les distances deux à deux entre les centres de gravité des différentes zones de l'objet peuvent alors être prises en compte, tout comme les différences angulaires entre chaque segment. Ceci correspond à une généralisation de la proposition développée dans [PRTM04], illustrée dans la figure 1.16 dans le cas où la vérité terrain est constituée de trois éléments :

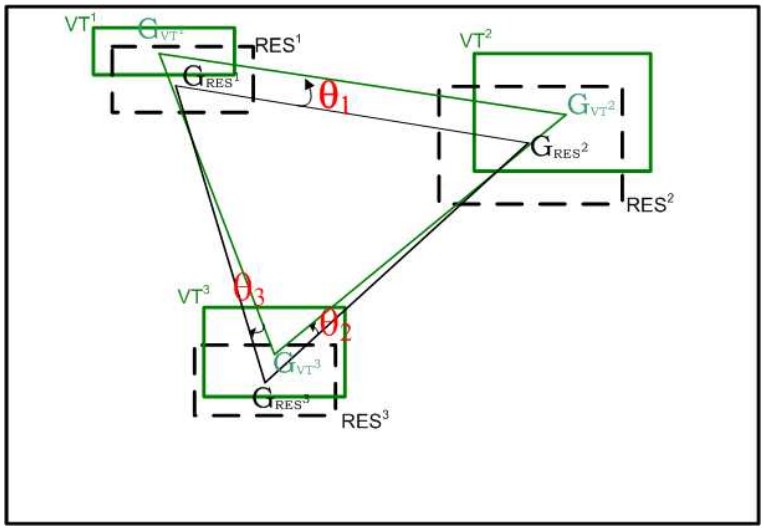


FIG. 1.16 – Les mesures d'évaluation dans le cas d'une vérité terrain constituée de trois éléments distincts : calcul des angles θ_i et des rapports $\frac{G_{VTi}G_{VTj}}{G_{RESi}G_{RESj}}$.

1.3.2 Pour le suivi

Suivre un objet consiste à regrouper toutes ses instances pour former un objet spatio-temporel. Dans la plupart des travaux de l'état de l'art, des mesures "*mixtes*" pour l'évaluation du suivi sont proposées, prenant en compte dans le même temps la qualité spatiale et temporelle des délimitations proposées des objets. Pour isoler des disfonctionnements propres à un module particulier (ici le suivi), il est préférable de baser son évaluation sur des critères lui étant les plus spécifiques possibles. Ainsi, pour ne pas faire d'amalgame avec l'évaluation du module de

détection, seuls des critères temporels seront pris en compte pour évaluer le module de suivi.

On notera par la suite $\mathcal{I}(VT_i) = \{VT_i^1 \dots VT_i^{N_{VT_i}}\}$, l'ensemble des N_{VT_i} instances (ou *séquence*) d'un objet i de la vérité terrain, et $\mathcal{I}_k^{Res}(VT_i)$ l'ensemble k des instances de l'objet i produit par le système. Les erreurs à prendre en compte sont alors :

- **la segmentation** : le système génère plusieurs ensembles d'instances pour un même objet i de la vérité terrain
- **la fusion** : plusieurs objets différents sont réunis dans un même ensemble d'instances $\mathcal{I}_k^{Res}(VT_i)$,
- **les cas mixtes** : un même objet i appartient à plusieurs séquences et certaines de ces séquences contiennent des instances d'autres objets.
- **l'oubli** : une instance de l'objet i n'est associée à aucune séquence produite par le système.

La mesure d'évaluation consiste alors à comptabiliser ces différents cas. Le but est de pénaliser le module de suivi en fonction du nombre et de la "qualité" des erreurs auxquelles il aboutit. Le comptage des différentes erreurs repose sur les données conservées dans la vérité terrain. Il est important de rappeler ici que cette vérité terrain est constituée relativement aux résultats du module de détection après son adaptation, puisque la méthodologie d'adaptation adoptée est séquentielle. Cette remarque s'applique d'ailleurs aussi à l'évaluation des modules d'amélioration et de reconnaissance.

Etant donné un objet de la vérité terrain VT_i, toutes les séquences contenant au moins une instance de l'objet sont regroupées : $\{\mathcal{I}_k^{Res}\}_{k=1\dots N_{seq}}$. Pour chacune de ces séquences, les indices suivants sont ensuite calculés :

1. $I_{oubli} = 1 - \frac{N_{oubli}}{N_{VT_i}}$
2. $I_{segm} = \frac{1}{N_{seq}}$
3. $I_{fusion} = MOY_{k\in\{k=1\dots N_{seq}\}} \left(\frac{(N^k - N_{diff}^k)}{N^k} \times \frac{N_{VT_i}^k}{N^k} \right)$

Quelques précisions quant aux notations : N_{oubli} correspond au nombre d'instances de l'objet VT_i absentes de chacune des séquences $\mathcal{I}_k^{Res}$; N_{diff}^k désigne le nombre d'objets différents contenus dans la séquence $\mathcal{I}_k^{Res}$, dont le cardinal est noté N^k et enfin $N_{VT_i}^k$ correspond au nombre d'instances de l'objet VT_i contenus dans cette même séquence. Les deux premiers indices rendent compte de la qualité en termes d'oublis et de segmentation. I_{fusion} quantifie quant à lui le degré de fusion observé. Les cas mixtes sont par ailleurs aussi pris en compte dans ce dernier indice.

La mesure adoptée pour le suivi correspond alors au vecteur formé par ces trois indices :

$$\mathcal{M}^{suivi}(Vt_i) = \begin{pmatrix} I_{oubli} \\ I_{segm} \\ I_{fusion} \end{pmatrix} \tag{1.5}$$

La figure 1.17 donne quelques exemples permettant d'illustrer les différentes situations qui viennent d'être évoquées :

1.3.3 Pour l'amélioration

L'évaluation du module d'amélioration consiste à évaluer l'influence de celui-ci sur les résultats du module de reconnaissance. Sa mesure d'évaluation est donc liée à celle utilisée pour le

(a) Le cas idéal : toutes les instances de l'objet sont correctement associées en une unique séquence

(b) Un cas de suivi avec oublis : $I_{oubli} = 0.7$, $I_{segm} = 1$ et $I_{fusion} = 1$

(c) Un cas de segmentation avec oublis : $I_{oubli} = 0.9$, $I_{segm} = 0.5$ et $I_{fusion} = 1$

(d) Le cas le plus complexe : oublis, segmentation et fusion : $I_{oubli} = 0.9$, $I_{segm} = 0.66$ et $I_{fusion} = Moy(0.25, 0.36) = 0.30$

FIG. 1.17 – Exemples d'application de l'évaluation du module de suivi

module de reconnaissance.
Soit seq_i une séquence d'objets présentée en entrée du module d'amélioration. La sortie de ce dernier peut alors prendre deux formes et le calcul de la mesure d'évaluation $\mathcal{M}^{amelioration}$ dépend alors de la forme adoptée. Dans le cas où cette sortie consiste en une nouvelle séquence d'objets seq_i^{am}, dans laquelle chaque objet correspond à une amélioration d'un objet de la séquence initiale (*amélioration simple*), la mesure d'évaluation repose sur la comparaison deux à deux entre les résultats de la reconnaissance sur les objets de la séquence initiale et les résultats de la reconnaissance sur les mêmes objets "améliorés". Plus précisément, la mesure d'évaluation consiste en une moyenne de la différence entre ces différentes mesures d'évaluation de la reconnaissance.
Dans le cas où la sortie du module d'amélioration consiste en un unique objet obj^{am}, conçu à partir de toutes les instances d'objets de la séquence initiale seq_i (*amélioration composée*), la mesure d'évaluation repose sur la comparaison entre la mesure d'évaluation de la reconnaissance obtenue sur ce nouvel objet et l'ensemble des mesures d'évaluation calculées sur l'ensemble des instances constituant la séquence initiale. Pour effectuer cette comparaison, la mesure d'évaluation du meilleur résultat de la reconnaissance produit sur les instances de la séquence initiale est choisie. Il convient de noter ici qu'il est possible, dans le cas particulier du traitement de l'objet texte, d'utiliser des méthodes statistiques plus poussées pour générer à partir de l'ensemble des résultats de la reconnaissance obtenus sur les objets de la séquence initiale une unique chaîne de caractère [Jol04].

Etant donné une séquence $seq_i = \{obj_k\}_k$ en entrée du module d'amélioration, la mesure d'évaluation adoptée est donc la suivante :

Dans le cas d'une amélioration simple

$$\mathcal{M}^{am}(seq_i) = moy_k \left(\frac{|\mathcal{M}^{reco}(obj_k) - \mathcal{M}^{reco}(obj_k^{am})|}{max(\mathcal{M}^{reco}(obj_k), \mathcal{M}^{reco}(obj_k^{am}))} \right)$$

Dans le cas d'une amélioration composée

$$\mathcal{M}^{am}(seq_i) = \frac{|\mathcal{M}^{reco}(obj^{am}) - max_k(\mathcal{M}^{reco}(obj_k))|}{max(\mathcal{M}^{reco}(obj_k), max_k(\mathcal{M}^{reco}(obj_k)))}$$

où $\mathcal{M}^{reco}(obj_k)$ correspond à la mesure d'évaluation de la tâche de reconnaissance pour l'objet k.

1.3.4 Pour la reconnaissance

La distinction est ici marquée entre l'objet texte, pour lequel la mesure de Levenstein est adoptée et les autres objets. Pour ces derniers, la reconnaissance (ou identificaton) consiste à associer, par comparaison, l'objet considéré à un objet déjà identifié dans une base de donnée. Dans ce cas, la mesure d'évaluation de la reconnaissance choisie est celle, déjà évoquée, et utilisée dans le cas des visages ou de l'iris, du *taux de reconnaissance au premier rang.*

1.4 La question de la construction des vérités terrains

1.4.1 Le modèle adopté

Dans le cadre des DSRO, étant donné le peu d'ambiguïté existant lors de l'annotation des documents, tous les modèles de données adoptés pour stocker la vérité terrain sont sensiblement les mêmes et incluent les informations suivantes :

1. la délimitation spatio-temporelle de l'objet (contours ou boîte englobante),
2. l'identification de l'objet.

Il est un point important à noter et pourtant souvent négligé, à savoir que l'évaluation se trouve grandement facilitée si les modèles de données utilisés pour stocker les résultats et les annotations sont les mêmes.
Particulièrement pratique dans le cadre de la sérialisation des objets (dans le sens de la programmation orientée objet), le langage XML est adopté comme le standard dans de nombreux travaux, et le sera ici aussi. La figure 1.18 présente alors le schéma de description défini pour stocker les vérités terrains. Ce schéma correspond au caneva des descriptions générées en sérialisant un ensemble de classes *ad hoc* dont nous ne présentons pas ici par souci de clarté, les schémas UML.

Les objets peuvent être en mouvement. Les flux manipulés étant discrets (ce sont des ensembles d'images), ce mouvement peut être décomposé en autant de mouvements élémentaires que nécessaire. Chacun d'entre eux possède alors un temps de début et un temps de fin, et une identification unique lui est attribuée.
Le lien au document sur lequel est construite la vérité terrain est établi par les données stockées dans un *objet* (au sens de la programmation) *Media_Description*, l'ensemble des *objets* de ce type représentant les différents formats existants d'un même document (résolution, format d'encodage, etc). L'exemple d'une annotation pour un objet quelconque est donné ci-après.

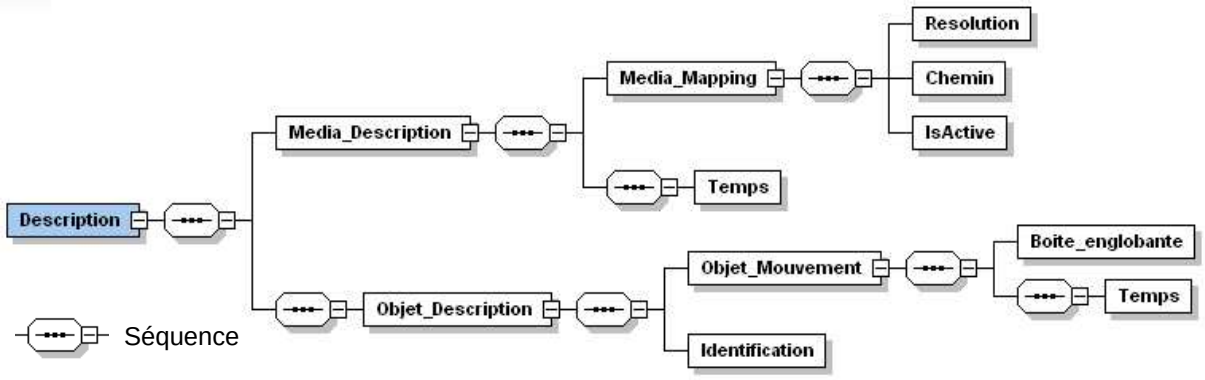


FIG. 1.18 – Le schéma de description pour stocker les vérités terrains

```
<?xml version="1.0" encoding="utf-8"?>
<Description xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Objets>
    <ObjetDescription>
      <identification> Objet1</identification>
      <Mouvements>
        <Objet_Mouvement>
          <TempsDébut>00:30:01</TempsDébut>
          <TempsFin>00:30:04</TempsFin>
          <Boite_englobante>
            <Location>
              <X>155</X>
              <Y>184</Y>
            </Location>
            <Size>
              <Width>172</Width>
              <Height>45</Height>
            </Size>
          </Boite_englobante>
        </ObjetMouvement>
        <ObjetMouvement>
        <TempsDébut>00:30:04</TempsDébut>
         <TempsFin>00:30:05</TempsFin>
          <Boite_englobante>
            <Location>
              <X>170</X>
              <Y>150</Y>
            </Location>
            <Size>
              <Width>135</Width>
              <Height>37</Height>
            </Size>
          </Boite_englobante>
        </ObjetMouvement>
      </Mouvements>
    </ObjetDescription>
                          .
                          .
                          .
      </Objets>
  <Media_Description>
    <durée>01:30:00</durée>
    <MediaMappings>
    <MediaMapping>
        <frameRate>25</frameRate>
        <Resolution>
          <Width>720</Width>
          <Height>576</Height>
```

```
        </Resolution>
          <string>ma_machine\ma_video.mpeg2</string>
        <name>video_haute_resolution</name>
        <isActive>true</isActive>
     </MediaMapping>
     <MediaMapping>
        <frameRate>25</frameRate>
        <Resolution>
          <Width>368</Width>
          <Height>288</Height>
        </Resolution>
          <string>ma_machine\ma_video.mpeg1</string>
        <name>video_basse_resolution</name>
        <isActive>false</isActive>
      </MediaMapping>
    </MediaMappings>
  </Media_Description>
  <TempsDébut>00:30:00</TempsDébut>
  <TempsFin>01:30:00</TempsFin>
</Description >
```

Cette vérité terrain décrit donc un document dont on dispose de deux formats (mpeg1 et mpeg2) et dans lequel apparaît un objet identifié comme étant "objet1". Ce document dure 1 heure et trente minutes et le traitement est appliqué sur le document au format mpeg2 pendant une heure, entre les dates 00 :30 :00 :00 et 00 :01 :30 :00. L'objet détecté suit un mouvement discrétisé en deux sous-mouvements. Sa boîte englobante se déplace ainsi de la zone repérée par le rectangle $R_1(155, 184, 135, 42)$ vers le rectangle $R_2(170, 150, 135, 37)$.

1.4.2 Quelques approches différentes

La construction des vérités terrains est une tâche particulièrement fastidieuse. Pour remédier à cela, certaines équipes proposent des solutions alternatives de construction permettant l'automatisation de cette tâche par consensus autour des résultats de différents systèmes [YP03]. Une autre solution consiste à anticiper les déplacements des objets en s'appuyant sur des heuristiques relatives, par exemple, à la stabilité de ceux-ci d'images en images. C'est le cas par exemple dans [BBTR04] où le système d'annotation, relatif au suivi de piétons propose par défaut dans chaque nouvelle image les zones délimitées par l'annotateur dans l'image précédente, arguant que les piétons apparaissent de façon quasi systématique dans plusieurs images successives. A charge alors à l'annotateur de supprimer la zone lors de la disparition du piéton. Les auteurs envisagent même de s'appuyer sur les résultats de systèmes de détection de mouvements pour limiter encore l'intervention de l'utilisateur.
Néanmoins, l'utilisation de systèmes automatiques en vue de limiter l'implication de l'utilisateur dans la création des vérités terrains n'apparaît pas comme une solution parfaitement viable, et ceci pour deux raisons :

1. le gain de temps obtenu par l'automatisation de la construction n'est pas si évident car l'utilisateur doit vérifier la cohérence de l'annotation proposée par le système, vérification pouvant être plus coûteuse que la création elle-même.
2. si l'utilisateur est complètement mis à l'écart du processus de construction, c'est toute la tâche d'évaluation qui doit être remise en cause puisqu'il devient impératif de prendre en compte l'incertitude de l'annotation dans la mesure adoptée.

Bien que laborieuse, la construction des vérités terrains se doit donc d'être entièrement manuelle. A charge ensuite de construire des applications réellement ergonomiques qui permettent de faciliter et d'accélérer l'annotation.

L'annotation étant naturellement sujette à erreur et à la subjectivité des annotateurs, il est par ailleurs possible de fusionner les résultats de différentes annotations ou encore d'établir un degré de confiance relatif aux annotations produites [SLB+05, BG97]. Etant donné la relative facilité (indépendamment de l'aspect fastidieux du travail) à construire les vérités terrains concernant les systèmes d'extraction d'objets, un unique annotateur est suffisant dans le cadre de notre étude. Pour autant, les règles d'annotation appliquées doivent être rigoureusement précisées pour éviter d'obtenir des résultats incohérents lors de l'évaluation.

1.4.3 Les outils nécessaires

Pour accélérer la tâche de construction des vérités terrains, des outils dédiés sont souvent développés, permettant de faciliter la navigation dans les documents, de délimiter précisément les zones englobantes des objets et de remplir tous les champs textuels nécessaires [LK03, DM00, YV98, HWZ04a]. Dans le cadre de nos expérimentations, notre propre outil a été développé et son fonctionnement sera exposé dans le premier chapitre de la partie 3.

1.5 Extraction des classes de comportements

L'évaluation des résultats d'un module produit un ensemble de vecteurs de performance qu'il convient de regrouper en classes de comportements homogènes. A l'issue de cette étape de clustering, les classes obtenues sont filtrées et seuls les comportements estimés "*insuffisants*" sont conservés en vue de l'optimisation. Une nouvelle représentation de ces derniers selon un ensemble de caractéristiques visuelles prédéterminé est enfin produite en vue de la mise en correspondance avec l'analyse du module considéré.
Les questions de la sélection de la méthode de clustering et de la méthode d'extraction des comportements insuffisants seront abordées dans ce chapitre où l'accent sera essentiellement porté sur la première de ces questions. Par ailleurs, nous présenterons dans un premier temps le cas particulier des classes de comportement correspondant aux fausses alarmes et aux oublis. Concernant le choix des caractéristiques visuelles utilisées pour produire la nouvelle représentation des comportements, le chapitre suivant présentera les contraintes mises en oeuvre.

1.5.1 Le cas particulier de la détection : comment gérer les fausses alarmes et les oublis

Dans le cas du module de détection, le seuillage des taux de recouvrement calculés aboutit immédiatement à deux classes de comportement particulières, les fausses alarmes et les oublis. Les vecteurs manipulés lors de l'étape de clustering représentent ainsi l'ensemble des comportements ne tombant dans aucune de ces deux catégories. Le choix du seuil utilisé pour extraire les fausses alarmes et les oublis se doit donc d'être judicieux pour éviter que les vecteurs de performance restant ne représentent uniquement des comportements "*parfaits*", pour lesquels l'incidence de l'adaptation serait par trop négligeable. Ce choix est donc le reflet de la précision avec laquelle l'utilisateur souhaite améliorer les résultats du système : plus le seuil est élevé, plus l'adaptation se résume au seul traitement des fausses alarmes et des oublis.
Or, si le traitement des oublis repose sur les mêmes principes que celui des autres comportements "*insuffisants*", celui des fausses alarmes met en jeu des rouages différents du fait de sa singularité. En effet, l'adaptation prend un sens différent pour les fausses alarmes : si dans le cas général (y compris pour les oublis), l'objectif est d'améliorer la qualité des résultats en précisant la

délimitation des objets relativement à la vérité terrain, dans le cas des fausses alarmes l'enjeu est de modifier le système de sorte qu'il ne détecte plus comme étant des objets certaines zones quelconques de l'image. Admettons ici que la même méthodologie d'adaptation soit appliquée aux fausses alarmes : à l'issue de leur représentation selon les caractéristiques visuelles (moyennant une étape de clustering selon ces dernières pour limiter l'hétérogénéité des éléments considérés), le module *responsable* de l'erreur constaté est identifié. Par la suite, les paramètres de ce module sont modifiés de sorte que ces fausses alarmes ne soient plus détectées par le module de détection. Selon le nombre de classes de fausses alarmes considérées, on obtient donc un ensemble de paramétrages permettant de s'assurer qu'aucun des éléments de ces classes ne sera plus détecté. C'est lors de la mise en application de ces paramétrages que cette méthodologie présente ses limitations. En effet, le scénario de l'usage des résultats de l'adaptation consiste à fusionner les résultats obtenus sur le document selon les différents paramétrages, pour obtenir un ensemble de résultats $\mathcal{R}^{opti}$. L'utilisation des paramétrages propres à la suppression des fausses alarmes devient alors problématique. En effet, la méthode la plus immédiate consiste à supprimer de l'ensemble $\mathcal{R}^{opti}$ les zones qui ne se retrouvent pas dans l'ensemble $\mathcal{R}^{FA}$ des résultats obtenus selon les paramétrages relatifs aux classes des fausses alarmes. Pour autant, il se peut aussi que certains résultats corrects de $\mathcal{R}^{opti}$ ne soient pas contenus dans $\mathcal{R}^{FA}$ puisqu'aucune connaissance n'est disponible sur la qualité des résultats de ce dernier ensemble : on sait uniquement qu'il ne contient pas de fausses alarmes. Dans ce cas, la méthode précité aboutit à la suppression de zones correctes (cf figure1.19).

La solution adoptée en réponse à ce problème consiste alors à traiter différemment le cas des fausses alarmes : chacune est tout d'abord représentée selon un ensemble de caractéristiques visuelles ; un clustering est ensuite effectué selon ces nouvelles représentations pour finalement créer des filtres basés sur ces caractéristiques, qui seront appliqués *in fine* sur l'ensemble $\mathcal{R}^{opti}$. La méthodologie de sélection des caractéristiques visuelles utilisées pour représenter les zones de l'image correspondant aux fausses alarmes n'est pas la même que celle mise en oeuvre pour la représentation des autres comportements. En effet, si la seconde repose sur des contraintes liées à la méthodologie d'analyse du fonctionnement des modules exposée dans le chapitre suivant, ces contraintes ne sont pas valides dans le cas des fausses alarmes. La méthode de sélection adéquate relative à ces comportements particuliers est ainsi beaucoup plus libre et sera exposée dans le premier chapitre de la partie 3, relatif aux expérimentations.
En conséquence, si les filtres produits peuvent aussi aboutir à la suppression de résultats corrects, leur contrôle est beaucoup plus simple que celui des résultats produits par les paramétrages relatifs aux classes de fausses alarmes puisqu'il est envisageable de modifier la nature des caractéristiques visuelles prises en compte en vue de limiter ce type d'erreur.

1.5.2 Choisir une méthode de clustering

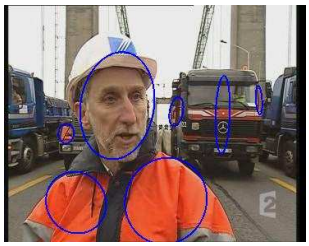
La contrainte principale à prendre en compte pour choisir la méthode de clustering adaptée à la classification des vecteurs de performance manipulés concerne l'ignorance du nombre de classes à considérer. Plusieurs des méthodologies de la littérature sont présentées ici avant de décrire le choix effectué.

1.5.2.1 Méthodes évolutionnistes

Le fonctionnement des algorithmes génétiques consiste à mimer les principes de l'évolution dans un contexte d'optimisation. Une population initiale de solutions potentielles est dans un premier temps constituée. La représentation des solutions adoptée dépend alors de la nature du

(a) Paramétrage 1

(b) Paramétrage 2 permettant de supprimer les fausses alarmes



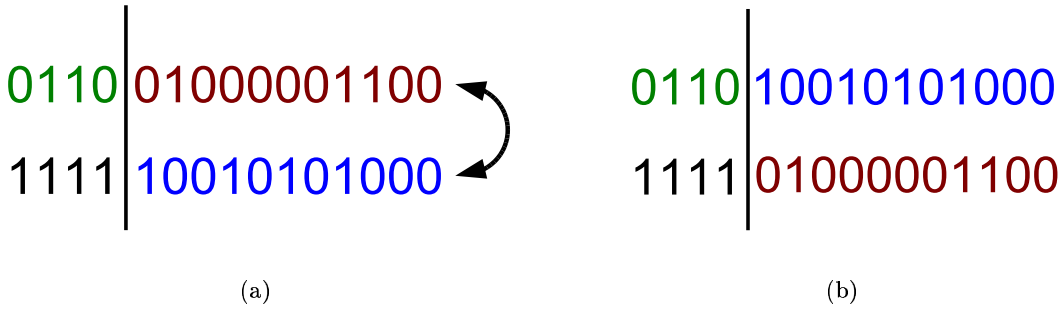
(c) Paramétrage 1

(d) Paramétrage 2 permettant de supprimer les fausses alarmes

FIG. 1.19 – Le système de détection de visages est appliqué avec deux paramétrages différents sur chacune des deux images. Les résultats de la seconde colonne correspondent au paramétrage le plus strict des deux pour lequel le nombre de fausses alarmes est limité (on assimilera ici ce paramétrage à celui pouvant être obtenu par l'analyse d'une classe de fausses-alarmes). On constate alors que la différence entre les résultats en termes de zones détectées ne permet pas, dans la seconde image, de pratiquer un filtrage correct puisque la zone délimitant le visage est elle aussi supprimée.

problème considéré (en général, celles-ci sont codées sous la forme d'une chaîne de bits ou de réels). Une mesure de la qualité de chaque solution, ou *mesure de fitness* est définie. Le déroulement temporel de l'algorithme repose alors sur un ensemble de *générations* dont les différentes étapes sont les suivantes :

1. Calcul des mesures de fitness des solutions de la génération N-1
2. Formation d'un *pool* de reproduction. Le choix des éléments s'effectue relativement à la mesure de fitness : les solutions sont dupliquées dans le pool proportionnellement à cette mesure. Le principe d'élitisme peut être appliqué à ce stade en préservant systématiquement les meilleures solutions de la génération N-1 dans la génération N.
3. Croisement (ou *crossover*) : des couples de solutions sont choisis au hasard dans le *pool* de reproduction. L'opérateur de croisement est alors appliqué à ce couple selon une certaine probabilité P_c. L'opérateur le plus courant est celui du croisement "*à 1-point*" : une position dans les chaînes considérées est choisie au hasard et les deux chaînes concernées "*échangent*" une partie de leur patrimoine comme illustré sur la figure 1.20
4. Mutation. Cette opération a pour but de favoriser le parcours le plus large possible de



(a) (b)

FIG. 1.20 – Croisement *à un point* entre deux chromosomes

l'espace des solutions. L'opérateur de mutation consiste à modifier la valeur de l'un des gènes d'un chromosome selon une probabilité P_m.

5. Le critère de fin repose soit sur un nombre maximal de générations, soit sur la proximité des solutions obtenues à l'issue de chaque génération (inversement proportionnelle à la probabilité d'obtenir des solutions différentes à la génération suivante), soit sur l'obtention d'une solution suffisante relativement à un critère portant sur la mesure de fitness. Dans le cas où ce critère n'est pas atteint, un nouveau cycle est entamé.

Il est possible d'associer les algorithmes génétiques à la recherche d'une partition optimale d'un ensemble de données [BM02, CGdLM03, Gre03]. Les trois méthodes proposées s'appuient sur une méthode de clustering particulière, l'aspect évolutionniste permettant ensuite de s'affranchir des contraintes inhérentes à ces méthodes.
Dans [BM02], la méthode de clustering est celle des k-means. En vue de déterminer automatiquement le nombre optimal de classes, une population est formée de chromosomes dont chaque gène représente la position d'un des centroids manipulés par les k-means. Le nombre de gènes non nuls de chaque chromosome, c'est à dire le nombre de classes à manipuler est choisi initialement au hasard. La mesure de fitness correspond dans ce cas à l'indice de Davies-Bouldin, mesurant la cohérence des classes produites par l'application de l'algorithme des k-means en fonction des centroids définis par chaque chromosome.
Dans [CGdLM03], l'algorithme de clustering sur lequel s'appuie la méthode repose sur l'élagage d'un arbre de recouvrement minimal ("*Minimum Spanning Tree*"). La même méthodologie que précédemment est appliquée : chaque chromosome référence les arcs devant être coupés. La mesure de fitness repose cette fois sur la mesure de Calinski et Harabasz.
Dans [Gre03], une méthode de clustering originale est présentée et associée à un algorithme génétique pour en améliorer les performances. Cette méthode est hiérarchique, incrémentale (une fois l'arbre construit, il est possible d'ajouter de nouvelles données sans reproduire l'ensemble des calculs) et non-supervisée. Le principe consiste à ajouter successivement tous les éléments à classer dans un ensemble. Son entropie est alors mise à jour et selon cette valeur, celui-ci peut être divisé en différents sous-ensembles. Un processus de fusion des sous-ensembles s'appuyant sur un critère similaire est aussi mis en oeuvre. Etant donné la dépendance de cette méthode à l'ordre dans lequel les éléments sont ajoutés, un algorithme génétique est utilisé, permettant de choisir l'ordre optimal d'ajout. La population initiale se compose de différents résultats de clustering obtenus en ajoutant les éléments dans un ordre différent. Les opérateurs de croisement et de mutation s'appuient sur l'aspect incrémental de la méthode de clustering. La mesure de fitness utilisée se fait écho de l'entropie des différentes solutions manipulées.

Les algorithmes basés sur les colonies de fourmis constituent une autre branche de ce domaine. Lorsqu'elles sont utilisées dans le cadre du clustering, ces méthodes exploitent la capacité des fourmis à disposer intelligemment les objets qu'elles rencontrent. La méthodologie proposée dans [AMS+03] s'inspire des travaux exposés dans [LF94]. En ce qui concerne la classification, le principe général des algorithmes à base de colonies de fourmis est le suivant : les données à classer sont projetées sur une grille 2D et les fourmis qui se déplacent sur cette grille peuvent déplacer des éléments en vue de former des tas (ou des zones si une case de la grille ne peut contenir qu'un unique élément) cohérents. Les différents paramètres de cette méthodologie générale sont ici détaillés dans le contexte de la méthode décrite dans [AMS+03].
La grille 2D considérée est toroïdale et les fourmis tout comme les éléments à classer y sont disposés aléatoirement. Le déplacement des fourmis est conditionné par une vitesse et un ensemble de probabilités concernant le choix de la direction. Par ailleurs, à chaque fourmi est associée une capacité, représentant le nombre d'éléments qu'elle peut transporter. A la différence de [LF94], chaque case de la grille peut accueillir un "tas" de plusieurs éléments. La cohérence d'un tel tas est évaluée relativement à différentes mesures concernant la distance entre les éléments de celui-ci. Les probabilités d'une fourmi à ramasser des éléments sur la grille ou à les y déposer sont alors définies relativement à ses mesures et à la capacité de celle-ci. La définition d'une donnée relative à la *patience* des fourmis les obligent à déposer les éléments qu'elles transportent au delà d'un certain nombre de déplacements. Enfin, les fourmis sont dotées d'une mémoire qui leur permet de mémoriser la nature des tas précécémment visités pour l'aider à choisir sa direction en vue de déposer les éléments qu'elles transportent. L'algorithme global alterne le travail des fourmis avec l'application d'un algorithme des K-means en vue d'affiner la partition produite.

1.5.2.2 Autres méthodes

Parmi les méthodes ne faisant pas appel au formalisme évolutionniste, on notera en premier lieu l'utilisation de méthodes de clustering hiérarchiques qui ne nécessitent pas le choix d'un nombre de classes *a priori*. Il est aussi possible d'itérer une méthode de clustering en faisant varier le nombre de classes désiré, puis à déterminer le nombre de classes optimal au regard d'un indice de cohérence du clustering produit (indice de Davies-Bouldin par exemple).
Une dernière méthode consiste à surestimer le nombre de classes puis à fusionner certaines des classes obtenues [FK96]. Dans cette étude, une version des "*Fuzzy k-means*" robuste au bruit et aux *outliers* (valeurs extrêmes s'écartant de la distribution) est d'abord proposée, dans laquelle deux ensembles de poids (au lieu d'un dans la méthode *classique* des *Fuzzy k-means*) sont associés aux éléments à classer. Le premier, contraint, représente pour un élément, le degré d'appartenance à chacune des classes. Le second est non-contraint et décrit la typicalité de chaque élément relativement à chaque classe. Lorsque le nombre de classes est inconnu, le principe est alors de surpartitionner les données, à la suite de quoi certains clusters sont fusionnés (le critère de comparaison utilisé repose sur l'analyse des degrés de typicalité et d'appartenance des éléments de chacun des deux clusters).

1.5.2.3 La méthode choisie

Les méthodes évolutionnistes paraissent sous un premier jour particulièrement attractives. Les algorithmes génétiques en l'occurence reposent sur une solide base théorique exposée dans [Whi94]. Pour autant, bien que leur implémentation ne pose pas de problèmes particuliers, leur

mise en application s'avère particulièrement difficile, étant donné le nombre de paramètres qu'il convient de fixer, résumés ci-dessous :

1. Taille de la population initiale
2. Mesure de fitness
3. Probabilité de croisement P_c
4. Probabilité de mutation P_m
5. Critère de fin

Il s'avère en outre que certains de ces paramètres sont corrélés, comme le montre l'équation 1.6 proposée par Holland [Hol75], établissant l'évolution de la population des solutions comprises dans un hyperplan particulier de l'espace de recherche, entre deux générations (t et t+1) :

$$P(H,t+1) > P(H,t)\frac{f(H,t)}{\overline{f}}\left[1 - P_c\frac{\Delta(H)}{L-1}(1 - P(H,t)\frac{f(H,t)}{\overline{f}}\right](1-P_m)^{o(H)} \qquad (1.6)$$

où P(H,t) désigne le nombre d'éléments de la population appartenant à l'hyperplan H lors de la génération t, f(H,t) correspond à la mesure de fitness de cet hyperplan, $\overline{f}$ désigne la mesure de fitnesse moyenne des éléments de la population totale, $\Delta(H)$ représente la longueur de définition de l'hyperplan H (la distance entre les deux bits fixés les plus éloignés de l'hyperplan, qui quantifie la probabilité que de dernier soit affecté par la recombinaison du *crossover* ($\Delta(*1** *011*10**) = 8$) et L la longueur d'un chromosome (la longueur d'encodage du problème). Enfin, o(H) représente l'ordre de l'hyperplan, c'est à dire dans une formalisation binaire, le nombre de bits fixés à 0 ou 1 dans sa représentation ($o(1**01**) = 3$). Cette formule montre qu'il est impératif d'étudier attentivement la question du paramétrage afin de garantir une évolution correcte des hyperplans contenant les solutions effectives au problème abordé.
Cette importance du paramétrage a d'ailleurs été expérimentée, puisque deux des algorithmes de clustering génétiques présentés dans cette partie ont été réimplémentés durant notre étude ([BM02, Gre03]) et le système [BM02] a de plus été testé sur des ensembles déjà labélisés de données [NHBM98]. Les résultats obtenus révèlent alors une sensibilité certaine à la variation de certains paramètres, illustrée dans la figure 1.21 dans le cadre du choix du nombre d'itérations maximal (critère de fin).

Une étude plus poussée de la dépendance des algorithmes génétiques relativement à leurs paramètres est par ailleurs proposée dans [DM02]. Cette dépendance donne lieu à une littérature fertile, dont l'aboutissement consiste à prendre en compte des paramètres auto-adaptatifs [Gom04].
En sus de ces considérations, il convient de remarquer que la taille de la population joue un rôle particulièrement déterminant dans la perspective d'une application effective des algorithmes génétiques. Les travaux fondateurs d'Holland [Hol75] ont montré que le parcours de l'espace de recherche était proportionnel au cube de la taille de la population. La taille de la population nécessaire à la détermination du nombre de classes optimal se révèle alors bien trop importante en termes de temps de calcul et constitue un nouveau frein à l'usage des algorithmes génétiques. Enfin, les algorithmes à base de colonies de fourmis souffrent des mêmes inconvénients relativement à leur paramétrage quio s'avère lui aussi complexe. Ces méthodes n'ont donc pas non plus été retenues pour être appliquées au clustering des vecteurs de performances.

La méthode de clustering choisie se doit de limiter l'intervention de l'utilisateur en ce qui concerne son paramétrage. Dans l'idéal, la méthode devra être indépendante :

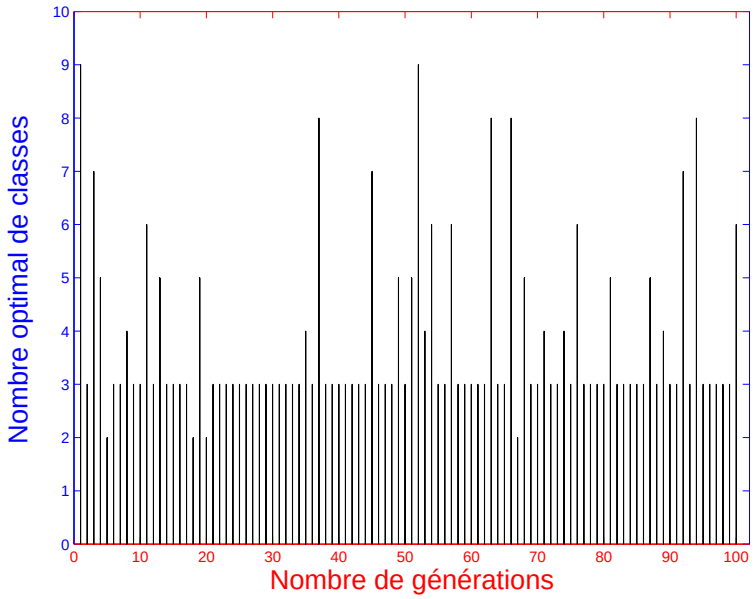


FIG. 1.21 – Dépendance du nombre de classes optimal en fonction du nombre de générations maximal autorisé

- aux mesures d'évaluation choisies,
- au corpus d'adaptation,
- à la distribution des vecteurs de performances,
- etc.

Pour ces différentes raisons, la méthode choisie propose pour ce faire une maîtrise effective de ses paramètres, se basant sur des algorithmes de clustering éprouvés (dont l'intérêt ne se dément pas, comme le montre la figure 1.22) : les k-means et un algorithme *classique* de clustering hiérarchique dit de "*linkage*" [DHS01].

Le principe de la méthode est alors très simple : choisir le meilleur résultat de clustering (selon un certain indice de cohérence) parmi un ensemble de résultats produits selon différents paramétrages des deux méthodes précitées (par compétition). Dans un premier temps, un ensemble de représentations hiérarchiques des données (des arbres) sont produites en appliquant l'algorithme de *linkage* selon différents paramétrages. La meilleure représentation est alors choisie selon la mesure de Cophenet (ou mesure de corrélation de Pearson) décrite dans l'équation 1.7 :

$$c = \frac{\sum_{i<j}(Y_{ij} - y)(Z_{ij} - z)}{\sqrt{\sum_{i<j}(Y_{ij} - y)^2 \sum_{i<j}(Z_{ij} - z)^2}} \tag{1.7}$$

où Y_{ij} correspond à la distance entre les données i et j, telle que calculée initialement avant de lancer le clustering hiérarchique et Z_{ij} désigne la distance entre les clusters auxquels appartiennent ces mêmes données. Enfin, y et z correspondent aux moyennes de Y_{ij} et Z_{ij} respectivement.
Dans le même temps, le nombre de classes optimal est déterminé en s'appuyant sur une application itérative de l'algorithme des k-means. Un résultat optimal de clustering est ensuite produit en faisant varier la position initiale des centroids des classes. Enfin, la meilleure représentation hiérarchique est coupée de sorte d'obtenir ce même nombre de classes et un indice de cohérence

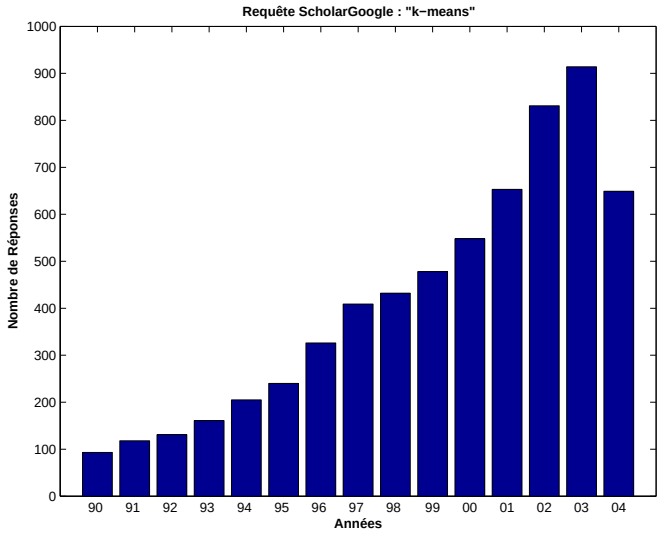


FIG. 1.22 – Réponse d'une requête dans Google Scholar, illustrant la popularité de la méthode des k-means

permet finalement de choisir, entre le résultat produit par les k-means et celui produit par la méthode hiérarchique.
Les paramètres pris en compte relèvent de l'algorithme hiérarchique lui-même mais aussi de la *préparation* des données :

1. Mode de normalisation des données,
2. Mise en oeuvre (ou non) d'une réduction par ACP des données,
3. Paramètres des méthodes hiérarchiques :
   (a) Distance utilisée pour mesurer l'écart entre les données,
   (b) Distance d'appartenance à un cluster (*élément-cluster* ou *cluster-cluster*),
   (c) Critère de validation des clusters obtenus.

   Détaillons ici les différentes alternatives envisagées :

1. **Normalisations** : normalisation $\sigma - \mu$ (ou $\mu - \sigma$) (la distribution des données est modélisée par une gaussienne) ou normalisation min-max qui consiste simplement à ramener la plage de chaque caractéristique entre 0 et 1 depuis sa plage de variation initiale entre *min* et *max*.
2. **Distances** : L_1, L_2, L_3 et L_∞
3. **Distances entre clusters** : $d_{min}(\mathcal{C}_1, \mathcal{C}_2) = min_{(c_i \in \mathcal{C}_1, c_j \in \mathcal{C}_2)}(\|c_i - c_j\|_{L_2})$, $d_{max}(\mathcal{C}_1, \mathcal{C}_2) = max_{(c_i \in \mathcal{C}_1, c_j \in \mathcal{C}_2)}(\|c_i - c_j\|_{L_2})$et $d_{moy}(\mathcal{C}_1, \mathcal{C}_2) = \|\overline{C}_1 - \overline{C}_2\|_{L_2}$
4. **Critère de validation**. L'objectif d'un algorithme de clustering étant, conjointement, de minimiser la dispersion intra-classe et de maximiser la dispersion inter-classe, la majorité des critères de validation prennent en compte ces deux quantités. Plusieurs indices sont alors définis à partir de ces notions de dispersion et l'objectif. Le critère choisi dans cette étude est celui, *classique*, de Davies-Bouldin [DB79] :

$$DB(U) = \frac{1}{c} \sum_{i=1}^{c} max_{i \neq j} \left\{ \frac{\Delta(X_i) + \Delta(X_j)}{\delta(X_i, Xj)} \right\}$$

où les X_i désignent les différents classes obtenues ; $\Delta(X_i)$ la dispersion intraclasse et $\delta(X_i, X_j)$ la dispersion interclasse. Il est à noter qu'il existe de nombreux autres critères de validation des résultats de la classification dont les indices de Fisher [Fis36], de Dunn [Rou87] ou encore la méthode de *silhouette* [Dun74], pour ne citer que les plus courants.

La figure 1.23 résume alors la méthodologie utilisée :

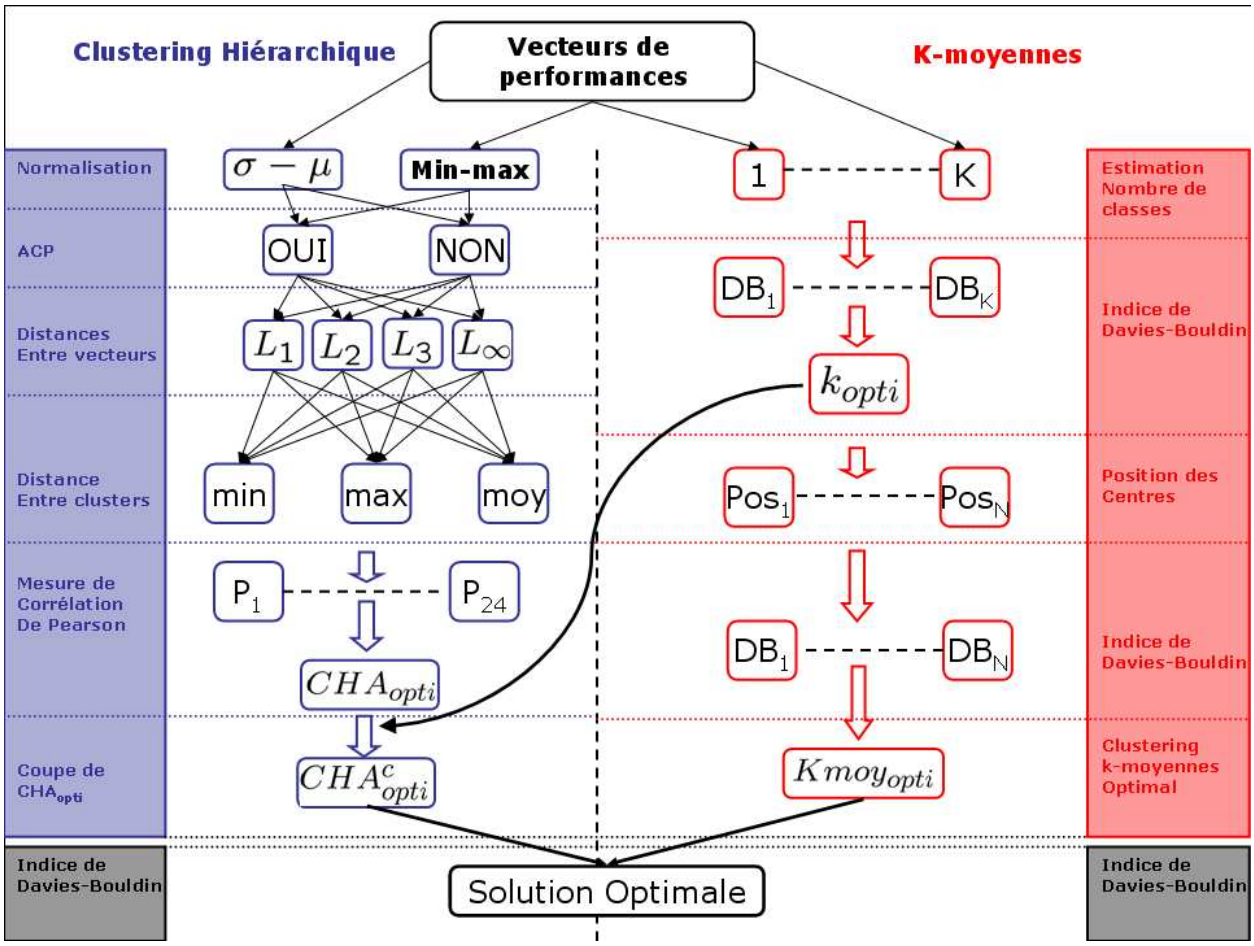


FIG. 1.23 – Le schéma global de la méthode de clustering utilisée

Il est à remarquer que cette méthodologie est modulaire : la prise en compte d'autres paramètres, en utilisant par exemple un processus de fusion (par la théorie de l'incertitude de Dempster-Schaeffer) de différents critères de validation est notamment envisageable. Le choix effectué ici ne se veut en aucune façon être le reflet d'une solution idéale et figée au problème du clustering des vecteurs de performances ; la question de l'extension de la méthodologie générale d'adaptation proposée dans cette étude à d'autres technologies potentiellement plus efficaces sera ainsi détaillée dans la conclusion de ce document.

1.5.2.4 Extraction des comportements insuffisants

Une fois les différentes classes extraites relativement à la méthode de clustering que nous venons de décrire, l'objectif est d'isoler les classes correspondant à des comportements dits "*insuffisants*", classes sur lesquelles l'analyse dite de "diagnostic de responsabilité" sera par la suite

appliquée.
La méthode utilisée ici consiste alors simplement à évaluer la qualité des performances de chaque classe relativement à la norme du vecteur correspondant au centroid de celle-ci. Selon la valeur de cette norme, la classe est alors considérée comme relevant d'un comportement insuffisant ou non.

1.6 Conclusion

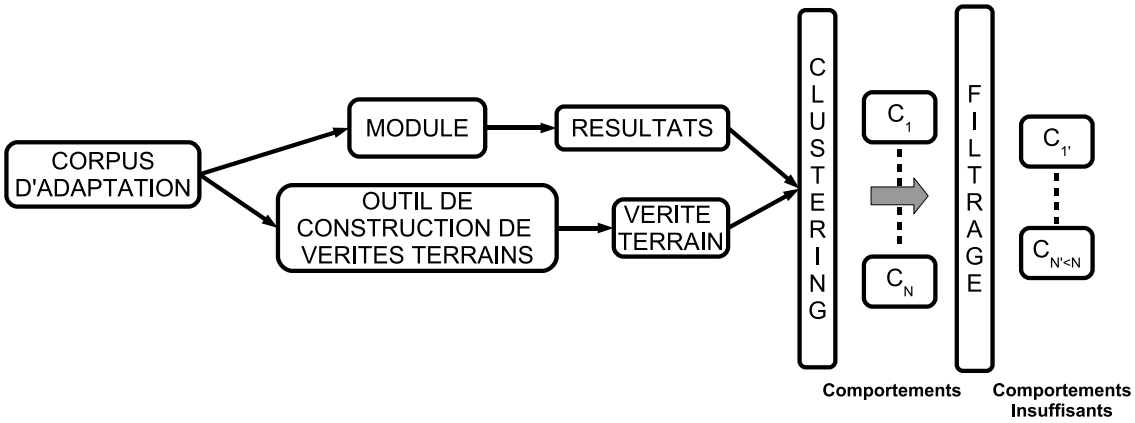


FIG. 1.24 – Le schéma global de l'analyse des comportements d'un module

La méthode d'extraction des comportements (cf figure 1.24), bien que très simple, constitue un des apports de notre méthodologie d'adaptation. Généralement, les résultats d'un système sont analysés relativement à différentes classes d'images choisies par l'utilisateur. Le principe adopté ici est de ne prendre aucun *a priori* sur le mode de fonctionnement du système, en préférant analyser des vecteurs de performances calculés en comparant les résultats du système à une vérité terrain. Une autre idée forte est que la constitution des seules classes des oublis et des fausses alarmes ne suffit pas : si la réduction de ces erreurs constitue un objectif important et souvant poursuivi dans la littérature, il est rare que les résultats "intermédiaires" soient considérés en vue d'être améliorés.
Par ailleurs, de nouvelles mesures d'évaluation des différents modules constituant un système DSRO ont été proposées tout comme une méthode de clustering des vecteurs de performance leur étant associés. Ces différents points sont bien sûr importants mais constituent une orientation "*technologique*" qu'il sera facile de modifier à l'avenir lorsqu'il sera envisagé d'appliquer la méthodologie à des systèmes différents.

A l'issue de l'analyse des comportements, c'est à dire une fois les comportements insuffisants extraits, la seconde étape de la méthodologie a pour objectif de déterminer pour chaque comportement, quel est le module de niveau inférieur responsable de l'erreur constatée. Le chapitre suivant portera ainsi sur ce point en détaillant chacune des deux étapes mises en jeu dans le cadre du *diagnostic de responsabilité*, à savoir l'adoption d'une représentation des comportements selon un ensemble de caractéristiques visuelles, ainsi que l'analyse des performances des modules de niveau inférieur relativement à ces caractéristiques.

2

Analyse du module et diagnostic de responsabilité

Sommaire

A l'issue de l'analyse des comportements d'un module M, les comportements insuffisants sont extraits. Ce module M peut être appréhendé comme une séquence de modules de *niveau inférieur*. Partant du principe qu'il existe, pour un comportement insuffisant donné, un module de niveau inférieur dont l'incidence sur le résultat final est telle que ce dernier peut être considéré comme responsable de l'échec constaté, l'enjeu consiste alors à rechercher ce module et à focaliser l'adaptation sur celui-ci. L'image d'une chaîne de montage est la plus parlante pour éclaircir nos objectifs. Admettons que cette chaîne soit constituée de plusieurs machines différentes. Elle produit des pièces mécaniques inspectées en sortie par des personnes spécialisées qui isolent les pièces défectueuses et les trient selon leurs défauts. La personne en charge de la chaîne doit limiter le plus possible de telles erreurs. Pour supprimer chaque type de défectuosité, cette personne n'inspecte pas l'ensemble des machines qui compose la chaîne : l'analyse de la forme des défauts constatés sur les pièces lui permet de reconnaître la machine responsable pour ensuite affiner ses réglages ou la réparer. Notre problématique pour les systèmes de vision consiste alors à suppléer au travail d'un tel expert en s'appuyant sur l'analyse des comportements pour produire un diagnostic en vue de limiter l'optimisation à un unique module composant le système.

Une méthodologie immédiate de recherche de la responsabilité consiste à évaluer successivement les résultats des différents modules de niveau inférieur mis en jeu. Néanmoins, l'inexistence de vérités terrains permettant une telle évaluation nécessite d'avoir recours à des indices de plus bas niveau pour dresser ce diagnostic.
La solution proposée s'appuie alors sur deux étapes distinctes :

1. Construire une nouvelle représentation des comportements insuffisants basée sur un ensemble de caractéristiques prédéterminé,
2. S'appuyer sur cette représentation, couplée à une analyse de la variabilité des performances des différents modules de niveau inférieur, pour établir le diagnostic de responsabilité.

Si la méthode de détermination du module responsable s'articule nécessairement autour de la succession de ces deux étapes, deux mises en application différentes ont été expérimentées au cours de ces travaux et seront présentées subséquemment. La première de ces deux méthodologies repose sur la quantification de la sensibilité des modules de niveaux inférieurs aux caractéristiques visuelles choisies. La seconde méthodologie s'appuie sur l'analyse des courbes d'évolution des résultats des différents modules de niveau inférieur en fonction des caractéristiques, permettant ainsi de remédier expérimentalement au problème, précédemment soulevé, de l'inexistence d'une représentation analytique des systèmes de vision, source de leur non-prédictibilité. Plus précisément, l'analyse de ces courbes prend en compte les différences relatives de performances des différents modules de niveau inférieur dans les différentes plages de variation de chaque caractéristique, déterminées relativement aux valeurs constatées sur les objets du comportement considéré. En référence à la contrainte d'autonomie adoptée, on soulignera ici l'usage, dans ces deux méthodes, des seules connaissances disponibles : les entrées des modules, sous la forme de la représentation des objets constituant les comportements ; ainsi que les sorties de tous les modules de niveau inférieur.

Une partie sera consacrée à la spécification de chacune de ces deux méthodes. Par ailleurs, nous verrons qu'elles nécessitent toutes deux de constituer un ensemble de caractéristiques visuelles auxquelles sont associées des bases d'objets dont les variations de chaque caractéristique sont contrôlées. Le choix des caractéristiques et la construction de ces bases donneront ainsi lieu à une dernière partie développant les choix effectués concernant ce point précis.

2.1 Une méthodologie basée sur l'analyse de la sensibilité

2.1.1 Principe et fonctionnement

Les informations disponibles pour pratiquer le diagnostic de responsabilité sont limitées. Une des informations exploitables est alors la nature des objets composant un comportement *insuffisant* relativement à un ensemble de caractéristiques visuelles. A partir de ces informations, il peut être envisagé d'établir le diagnostic en analysant le comportement de chaque module relativement à ces mêmes caractéristiques. Il n'existe pas de vérité terrain permettant d'évaluer la sortie de ces modules. En conséquence, une mesure quantitative de la qualité des résultats pour chaque valeur de caractéristique ne peut être obtenue. L'unique solution consiste alors à analyser les variations des résultats obtenus en fonction de ces caractéristiques en définissant une distance entre ceux-ci (typiquement une distance entre images). Cette analyse nous amène finalement à définir la notion de sensibilité :

Définition 12. *Un module est dit sensible à une caractéristique si l'application de celui-ci sur des objets pour lesquels cette caractéristique varie aboutit à des résultats différents.*

La quantification des variations des résultats observées est complexe : il est difficile de définir un seuil à partir duquel ces variations permettent de conclure à la sensibilité du module à la caractéristique prise en compte. En conséquence, une méthode originale d'évaluation de la sensibilité a été développée et sera détaillée dans la partie suivante.

2.1.2 Etude de la sensibilité

Etant donné un ensemble de caractéristiques visuelles $C^{visu} = \{C_i^{visu}\}_{i=1...N^{visu}}$, l'objectif est de quantifier la variabilité des résultats des différents modules de niveau inférieur composant le module considéré, relativement à la variation de ces caractéristiques.

2.1.2.1 Principe

La méthode développée dérive des travaux exposés dans [DMM01, Bur05], s'inspirant du principe d Helmholtz selon lequel " des structures remarquables dans une image peuvent être vues comme des exceptions à l'aléatoire". L'idée de ces travaux est alors de détecter les évènements saillants dans une image (limités à un ensemble de formes géométriques simples) en supposant qu'un évènement saillant se caractérise par une faible probabilité d'observation, relativement à une distribution des pixels dans l'image de la forme d'un bruit uniforme.

Cette approche peut alors être transposée à notre problématique en s'appuyant sur la définition suivante de la sensibilité :

Définition 13. *Un module est dit* ***sensible*** *aux variations d'une caractéristique f si ses résultats varient significativement en fonction des valeurs prises par f. Les variations constatées sont considérées comme significatives si elles sont statistiquement équivalentes (ou supérieures) à celles obtenues par un module stochastique de même nature.*

Le problème clé est alors de définir clairement, pour chaque module de niveau inférieur composant la séquence du module considéré, un opérateur stochastique équivalent. Une première méthodologie consiste à appliquer strictement le principe exposé dans [DMM01]. Dans ce cas, la définition des modules stochastiques ne prend pas en compte la nature des modifications apportées par le module considéré (détection d'un contour, augmentation du contraste, ...). Seule la nature des données manipulées par celui-ci est considérée et le module stochastique équivalent

est alors défini selon une simple distribution uniforme. Tous les opérateurs travaillant dans des espaces identiques (niveaux de gris par exemple) se voient ainsi associer des opérateurs stochastiques identiques.

Prenons le cas d'un détecteur de contours travaillant sur des images en niveaux de gris, l'opérateur stochastique équivalent consiste alors à assigner à chaque pixel une valeur en niveau de gris, sachant que chaque valeur est équiprobable. Lorsque le module considéré crée des zones englobantes, l'opérateur stochastique associé propose en sortie un ensemble de zones englobantes dont la position ainsi que la taille sont choisies de manière aléatoire dans les plages déterminées par les dimensions de l'image.

2.1.2.2 Mise en Application

Considérons maintenant un module composé de trois modules de niveau inférieur distincts, dont la sensibilité à une caractéristique C_k^{visu} doit être étudiée ($m = m_{sys}^0 \succ m_{sys}^1 \succ m_{sys}^2$, où l'indice *sys* signifie que le module considéré correspond à sa version originelle et non à son équivalent stochastique m_{stoch}). Le comportement de tout module est évalué dans le contexte de la séquence à laquelle il appartient. C'est à dire que les modifications apportées par le module le précédant doivent être prises en compte. En conséquence, si l'on admet qu'une base d'images $DB_{C_k^{visu}}$ dédiée à la caractéristique visuelle k a été construite, l'étude de la sensibilité du module m_{sys}^1 impose la mise en application des étapes suivantes :

1. Application de l'ensemble de la séquence "$m_{sys}^0 \succ m_{sys}^1$" à chacune des images de $DB_{C_k^{visu}}$,
2. Application de la séquence stochastique "$m_{sys}^0 \succ m_{stoch}^1$" autant de fois que $DB_{C_k^{visu}}$ contient d'images,
3. Calcul de la variance des résultats obtenus dans les deux cas ,
4. Comparaison des variances obtenues selon un test d'hypothèse bilatéral.

Un des points à préciser concerne l'évaluation de la variance des résultats obtenus par l'une ou l'autre des deux séquences considérées. Le calcul de cette variance impose l'usage d'une distance, bien entendu fonction de l'espace des résultats manipulés : images couleurs, images en niveaux de gris, images noir et blanc, ensemble de zones englobantes, ... La distance choisie repose sur la norme de Frobenius, équivalent matriciel à la norme euclidienne pour les vecteurs :

$$|| A ||_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} \|a_{ij}\|^2}$$

Dans le cas des images en niveaux de gris ou en noir et blanc, la norme de Frobenius peut être utilisée directement. Dans le cas des images en couleurs, la distance utilisée dérive de la norme de Frobenius pour tenir compte de l'existence des différents canaux de couleurs (on donne ici l'exemple en RGB) :

$$|| A ||_F^{R,G,B} = \frac{|| A ||_F^R + || A ||_F^G + || A ||_F^B}{3}$$

La distance associée à la norme de Frobenius est par définition une *vraie* distance. Par construction, il est très facile de montrer que la norme construite autour des différents canaux est aussi une *vraie* distance. Grâce à cette distance, le calcul des variances des résultats obtenus devient immédiat.

Lorsque les résultats correspondent à des ensembles de zones englobantes, le principe est de construire une image moyenne de ces résultats, en assignant à chaque pixel un niveau de gris proportionnel au nombre de fois où il appartient à une zone englobante. Finalement, le calcul de la variance repose sur la distance de Frobenius entre les images produites.

En ce qui concerne le test bilatéral de comparaison entre les deux variances calculées, une table de Fisher-Snédécor permet de rejetter ou d'accepter l'hypothèse H_0 selon laquelle celles-ci sont comparables (on suppose ici les distributions mises en jeu normales).

Cette méthodologie est expérimentée sur les trois modules suivants : les détecteurs de contours de Sobel et de Canny et l'opérateur de réhaussement de contours. La sensibilité de ces opérateurs est étudiée relativement aux caractéristiques suivantes : bruit, luminosité et contraste. Deux images *classiques*, celle de Lena et celle du cameraman, sont utilisées. Trois bases sont alors construites en faisant varier sur chacune de ces images une unique caractéristique (dans le cas du bruit, la variation porte sur la valeur de l'écart type du filtre gaussien appliqué). La plage de variation de chaque caractéristique est choisie expérimentalement : la base relative au bruit contient 51 images, celle concernant la luminosité 1200 images et enfin, celle correspondant au contraste 993 images. Les images de la figure 2.1 donnent un exemple des images contenues dans chacune de ces bases (pour l'image de *Lena*).

Les trois opérateurs considérés sont alors appliqués sur chacune de ces bases, tout comme leur équivalent stochastique, défini selon les principes précédemment énoncés. Les variances des résultats produits sur les différentes bases sont calculées par la distance de Frobenius. Les tableaux 2.1 et 2.2 montrent alors les résultats de comparaison des variances, sous la forme de la quantité à considérer lors de l'application du test de Fisher-Snedecor.

	Canny	Sobel	Rehaussement
Bruit	20.25	26.13	108.12
Luminosité	7.98	4.27	**0.90**
Contraste	2.36	2.67	4.45

TAB. 2.1 – Valeurs de la quantité à considérer pour le test de Snedecor de comparaison de variances (*Lena*), la valeur en gras correspondant à une valeur contenue dans l'intervalle de confiance à 5% définie selon la table statistique de Fisher-Snédécor.

	Canny	Sobel	Rehaussement
Bruit	15.06	18.76	49.10
Luminosité	6.50	5.64	**0.23**
Contraste	5.36	7.62	4.70

TAB. 2.2 – Valeurs de la quantité à considérer pour le test de Snedecor de comparaison de variances (*cameraman*).

Les valeurs 0.9 et 0.23, obtenues par l'opérateur de réhaussement relativement à la luminosité appartiennent à la plage d'acceptation du test de Fisher-Snédécor à 95%. On conclut donc à la sensibilité de cet opérateur à cette caractéristique.
Pour autant, l'analyse visuelle des résultats des opérateurs de détection de contours sur ces bases met en évidence leur sensibilité au contraste comme illustré dans les images de la figure 2.2.

Ce relatif échec nous incite donc à envisager une autre solution pour établir la sensibilité d'un module, solution proposée dans la suite de cette partie.

2.1.2.3 Alternative

L'alternative envisagée repose sur une représentation plus précise du comportement de chaque module m_{sys} grâce à une estimation de la distribution de l'opérateur stochastique équivalent obtenue, en se référant à la nature de la modification effectuée par m_{sys}. La modélisation de cette distribution se base désormais sur la détermination de $P(X^+ = x|X^- = y)$, représentant la probabilité d'obtenir au pixel X, après application du module considéré, la valeur x, sachant que ce pixel avait, avant application, la valeur y.
Prenons pour exemple un opérateur d'accumulation en niveaux de gris sur une fenêtre de taille W. Dans ce cas, l'obtention de la probabilité précitée repose sur un calcul de dénombrement : trouver l'ensemble des combinaisons de W-1 éléments dont la somme est égale à x-y. On obtient alors :

$$P(X^+ = x|X^- = y) = C^{|x-y|}_{|x-y|+W-2}$$

Donnons quelques éléments de démonstration :
Soit S(n,m) le nombre de n-uplets $(x_1, \ldots, x_n)$ d'entiers naturels compris entre 0 et m tels que $x_1 + \ldots + x_n = m$ Pour cette démonstration, le lemme suivant est utilisé :

Lemme 1.

$$\sum_{k=0}^{m} C^{n-1}_{n-1+k} = \sum_{k=1}^{m} \{C^n_{n+k} - C^n_{n-1+k}\} + C^{n-1}_{n-1} = C^n_{n+m} - C^n_n + C^{n-1}_{n-1} = C^n_{n+m}$$

Procédons par récurrence :
- **n=1** Il est évident que S(1,m)=1
- **n=2** Le premier élément x_1 varie entre 0 et m et le second, x_2, a alors pour valeur $m - x_1$. On obtient donc S(2,m)=m+1
- **n=3** Il est facile de montrer que $S(3, m) = \sum_{x_1=0}^{m} S(2, m - x_1) = C^2_{m+2}$
- **n=4** De la même façon, en utilisant le lemme, on obtient $S(4, m) = C^3_{m+3}$

On suppose ensuite que $S(n, m) = C^m_{m+n-1}$ et on le démontre par récurrence en utilisant le lemme.

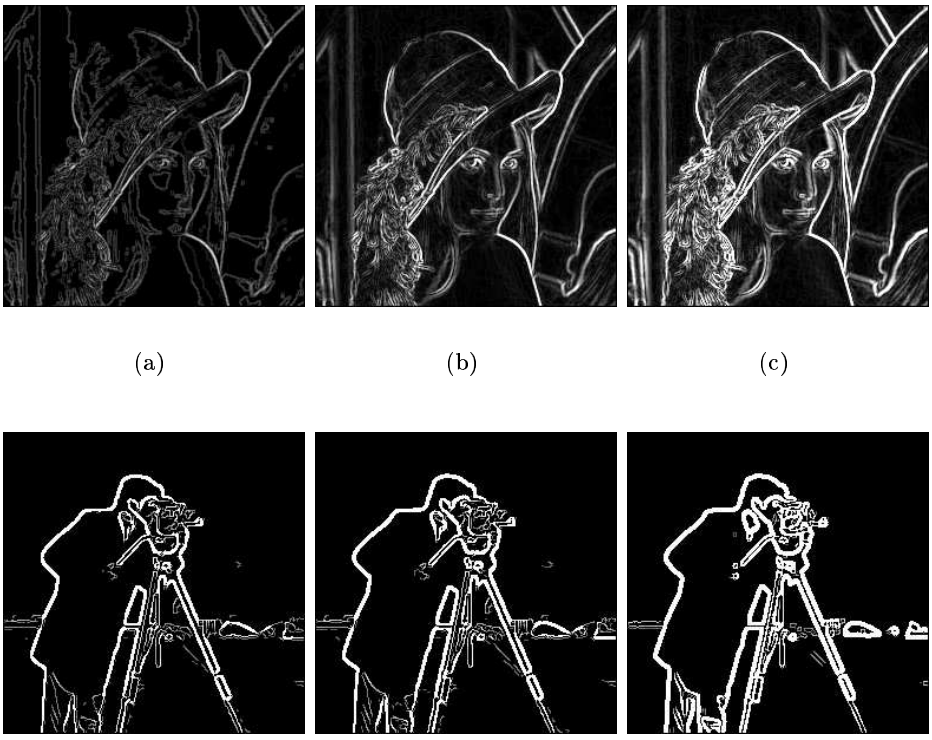
Cette courte démonstration a pour seul objectif de montrer toute la difficulté à établir des distributions en adéquation avec les modifications effectuées par le module *système* considéré. Déjà dans le cas d'un opérateur aussi simple que celui de l'accumulation, certains calculs de dénombrement sont nécessaires. Dans le cas d'opérateurs dont l'algorithmie associée s'avère plus "complexe", du point de vue du voisinage utilisé et(ou) de la fonctionnelle associée à ce voisinage (par exemple un gradient de sobel sur un voisinage 3×3), le calcul de dénombrement nécessaire devient beaucoup plus problématique.
Par ailleurs, il est à remarquer que ces calculs supposent toujours que la valeur prise par un pixel dépend d'une distribution uniforme (chaque valeur est équiprobable). Il conviendrait ainsi, non seulement de prendre en compte la nature des modifications opérées par le module considéré mais aussi celle de la relation spatiale entre les pixels. Une solution consiste alors à utiliser les champs de Markov aléatoires qui modélisent de façon probabiliste la dépendance de la valeur d'un pixel relativement aux valeurs des pixels d'un voisinage prédéfini. Malheureusement, une telle approche nécessite de déterminer auparavant un modèle paramétré de la distribution des valeurs, estimé ensuite en fonction des observations. Ce modèle est souvent supposé gaussien sans qu'aucune réelle justification de cet *a priori* ne soit donnée.

(a) $\sigma = 0.5$ (b) $\sigma = 1.75$ (c) $\sigma = 2.95$

(d) facteur = -0.600 (e) facteur = 0 (f) facteur = 0.599

(g) facteur = 0.04 (h) facteur = 2.515 (i) facteur = 4.995

FIG. 2.1 – Images extraites des trois bases concernant *Lena* : la première ligne correspond à la base dédiée au bruit, la seconde correspond à la base dédiée à la luminosité et la troisième à la base liée au contraste. σ correspond à la variance du bruit gaussien appliqué.



(a) (b) (c)

(d) (e) (f)

FIG. 2.2 – L'opérateur de Sobel (ligne 1) et l'opérateur de Canny (ligne 2) sont appliqués sur des images issues des bases relatives au contraste. L'analyse visuelle des résultats montre que ces opérateurs sont sensibles au contraste.

2.1.3 Conclusion

Les nombreuses difficultés rencontrées lors de la mise en application du principe de Helmholtz nous ont amené à abandonner cette première approche du diagnostic de responsabilité basée sur la sensibilité. La seconde méthode, présentée dans la partie suivante, propose alors une alternative plus simple au problème du diagnostic de responsabilité, basée sur l'analyse des courbes d'évolution des résultats des modules en fonction de l'ensemble de caractéristiques visuelles utilisé.

2.2 Une méthodologie "*analytique*"

2.2.1 Principe et fonctionnement

Dans un premier temps, les classes de comportements sont représentées par l'ensemble des plages de variation des caractéristiques visuelles. Chaque module de niveau inférieur est ensuite appliqué (selon le mode déjà présenté dans la partie 2.1.2.2) sur chacun des objets des bases construites relativement à ces mêmes caractéristiques. Cette fois-ci, l'enjeu n'est pas d'étudier la variation des résultats obtenus par chaque module de niveau inférieur mais plutôt d'établir une mesure de leur qualité relativement au résultat final attendu à la fin de la chaîne de traitement. Pour y parvenir, la sortie de chacun des modules est comparée, selon la distance de Frobenius, avec une vérité terrain correspondant à la sortie finale attendue du module cible de l'adaptation (cf figure 2.3).

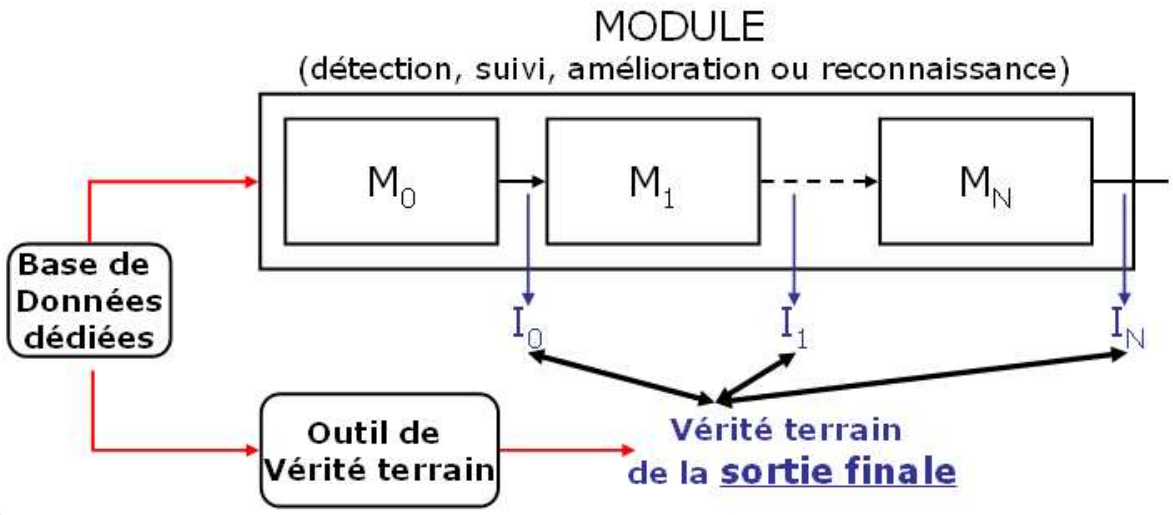


FIG. 2.3 – Les performances des modules de niveau inférieur sont évaluées relativement à la sortie finale du module

Une des contraintes d'application de la méthologie (évoquée dans le chapitre 3 de la partie 1) est que les sorties de chaque module de niveau inférieur considéré soient du même type que la sortie finale. Cette contrainte permet ici de faciliter la comparaison de ces sorties intermédiaires avec la sortie finale attendue (la vérité terrain). Dans le cas de la détection par exemple, cette vérité terrain est une image binaire dans laquelle l'objet est délimité par une zone blanche. Les différentes mesures obtenues permettent alors de construire les courbes qui serviront de base à

l'établissement du diagnostic de responsabilité.
La figure 2.4 (factice pour plus de clarté) illustre les courbes qu'il est possible d'obtenir relativement à une caractéristique C_i. Les variations relatives de ces courbes montrent un cas "*favorable*" pour lequel l'amélioration de la qualité des résultats au cours de la séquence est constante, quelle que soit la valeur de C_i (i.e : les quantités M2 - M1 et M1 - M0 sont constantes).

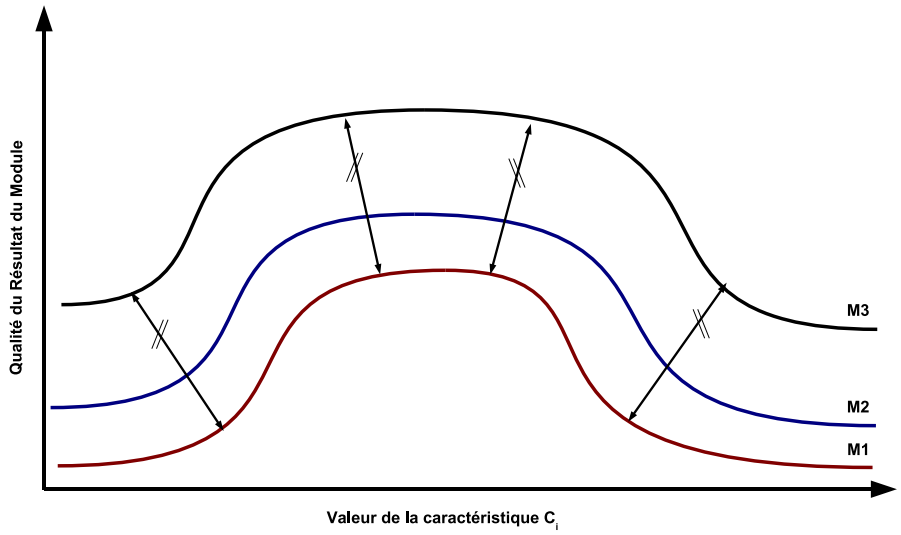


FIG. 2.4 – Un exemple de courbes représentant l'évolution de la qualité des résultats aux cours de la séquence du module pour différentes valeurs de la caractéristique C_i. M1, M2 et M3 désignent trois modules successifs dans la séquence considérée.

L'établissement du diagnostic de responsabilité repose alors sur le postulat suivant :

Postulat 2. *Chaque module de niveau inférieur entrant dans la composition de la séquence du module cible de l'adaptation, tend à améliorer les résultats obtenus par le module inférieur le précédant dans la séquence.*

Une première validation de ce postulat relève de son utilisation dans le contexte de la planification des systèmes de traitement d'images ([DBB00, DD98]). Dans ces travaux, le choix des opérateurs constituant la séquence de traitement est ainsi guidé par la capacité des opérateurs à faire évoluer les données transitant dans la séquence vers le résultat final attendu. Une seconde validation de l'adoption de ce postulat, expérimentale, sera par ailleurs présentée dans le chapitre 2 de la partie 3 de ce manuscrit, consacrée à l'application de la méthodologie d'adaptation à l'objet texte.
Le principe du diagnostic repose sur l'identification du module dont le comportement local (c'est à dire dans la plage de variation de chaque caractéristique mesurée sur les objets de la classe de comportement considérée) va le plus à l'encontre de ce postulat. Plus précisément, pour chaque caractéristique, l'analyse du comportement local de chaque module permet de lui assigner un indice de responsabilité. Un ensemble de diagnostics est par la suite produit par comparaison de ces indices, en considérant chaque caractéristique isolément (pour chaque caractéristique, le module obtenant l'indice de responsabilité le plus faible selon le mode de calcul détaillé par la suite, est considéré responsable). Finalement, les résultats de ces différents diagnostics sont fusionnés pour désigner le module responsable. Le schéma 2.5 résume la méthode de diagnostic utilisée.

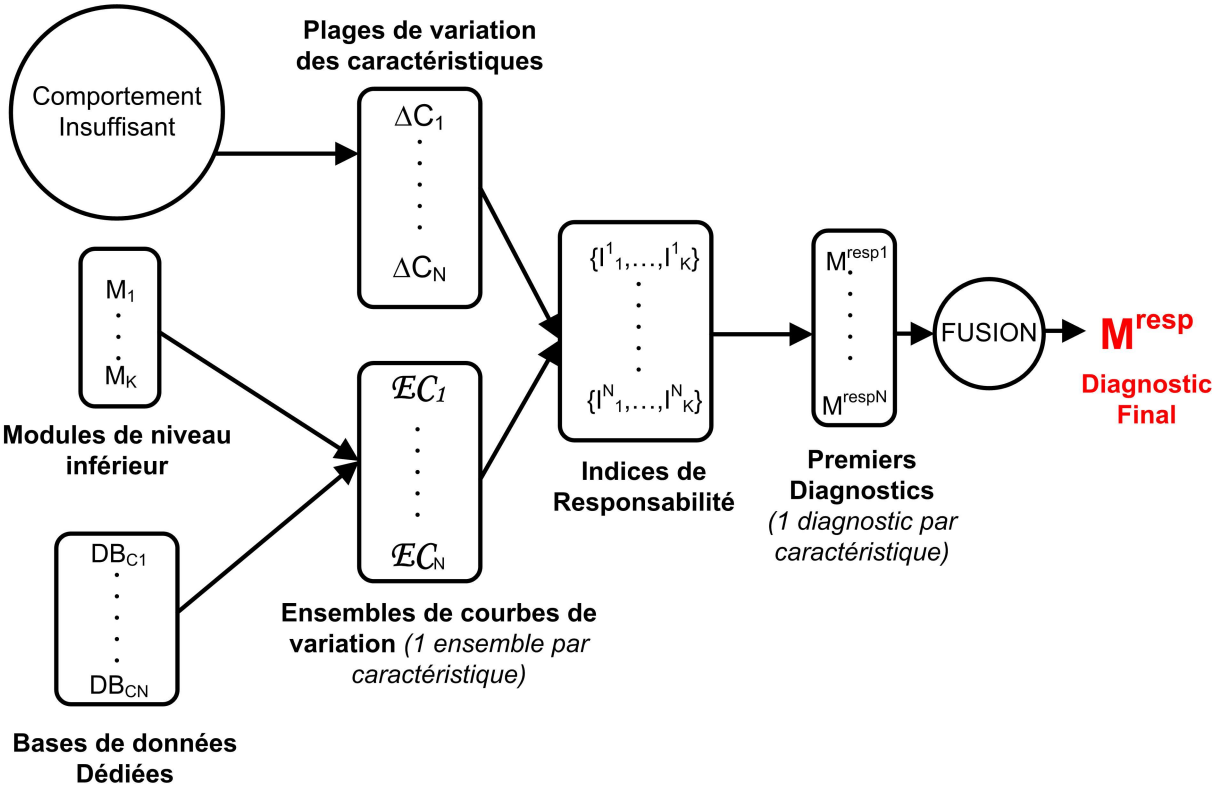


FIG. 2.5 – Schéma de la méthode de diagnostic

2.2.2 Calcul de l'indice de responsabilité

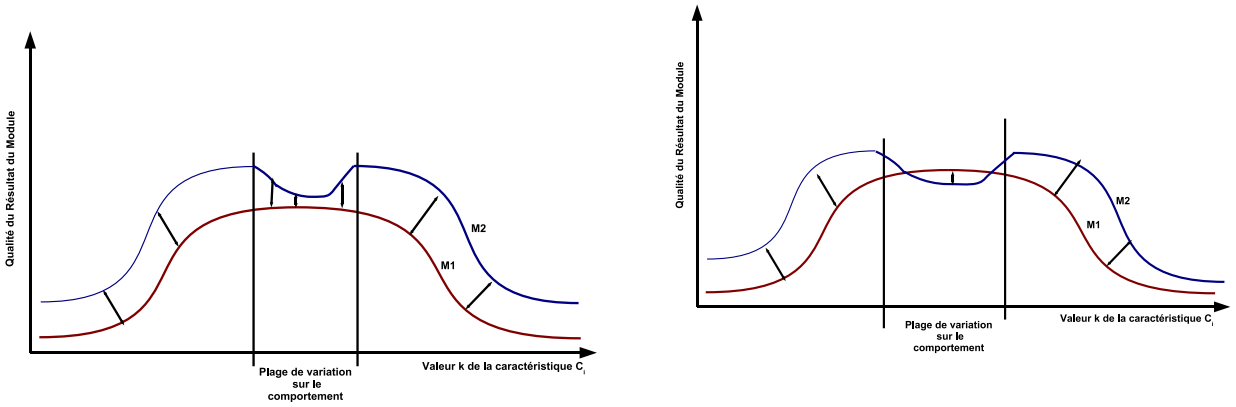
L'indice de responsabilité d'un module de niveau inférieur, pour une caractéristique donnée (c'est à dire sur la plage de variation de cette caractéristique, mesurée sur les éléments composant le comportement étudié), est établi en termes de dégradation locale des performances au regard de celles obtenues par le module de niveau inférieur le précédant dans la séquence. Cette dégradation peut prendre deux formes :

1. l'amélioration de la qualité des résultats constatée est "*exceptionnellement*" faible au regard de l'amélioration constatée sur la plage entière de valeurs de la caractéristique,
2. diminution locale "*exceptionnelle*" de la qualité des résultats.

Il est important de souligner ici l'aspect local de cet indice : la dégradation constatée se doit de refléter un comportement local exceptionnel du module de niveau inférieur par rapport au comportement observé sur l'ensemble des valeurs de la caractéristique. Les deux cas possibles de dégradation sont illustrés dans la figure 2.6.

La mesure de la dégradation repose alors sur l'évaluation des différences entre la courbe du module considéré et celle du module précédent sur la plage de la caractéristique considérée. Cette mesure est normalisée par la différence maximale constatée sur l'ensemble des plages de même largeur. Plus précisément, si on note C_i la caractéristique, l'indice de responsabilité $\mathcal{I}(M_j)$ attribué au module M_j est le suivant :

$$\mathcal{I}(M_j) = argmin_{(x,y)} \left(\frac{\int_{C_{min}}^{C_{max}} (M_j - M_{j-1})\, dC_i}{\int_x^y (M_j - M_{j-1})\, dC_i} \right) \; avec\; y - x = C_{max} - C_{min},\; \forall j \geq 1 \quad (2.1)$$



(a) La dégradation correspond ici à une amélioration insuffisante

(b) La dégradation correspond ici à diminution de la qualité des résultats

FIG. 2.6 – Les deux cas de dégradation considérés

où C_{min} et C_{max} délimitent la plage de variation de C_i sur le comportement.
Pour le premier module M_0, la formule est ajustée en quantifiant uniquement la singularité du comportement observé sur la plage $[C_{min}, C_{max}]$, selon l'équation suivante :

$$\mathcal{I}(M_0) = argmin_{x,y} \left(\frac{\int_{C_{min}}^{C_{max}} M_0 \, dC_i}{\int_x^y M_0 \, dC_i} \right) \; avec \; y - x = C_{max} - C_{min} \tag{2.2}$$

L'indice de responsabilité varie sur l'intervalle $]-\infty, 1]$. Pour chaque caractéristique, le module obtenant l'indice de responsabilité minimal est considéré comme *responsable*. On obtient donc dans un premier temps pour chaque comportement, un module de niveau inférieur *responsable* par caractéristique. Concernant la fusion de ces diagnostics, un vote à la majorité est effectué : le module désigné comme responsable dans un maximum de cas est choisi. Lorsque ce vote aboutit à une ambiguïté, c'est l'ordre de mobilisation des modules (de niveau inférieur) responsables qui détermine le diagnostic final : le premier module à être appliqué est choisi. Par ailleurs, l'indice de responsabilité est fixé à 1 dans deux situations :

1. lorsque la plage $[C_{min}, C_{max}]$ correspond à la plage sur laquelle les performances du module sont maximales au regard de la plage totale de la caractéristique,
2. lorsque les résultats du module M_j sont égaux à ceux du module précédent M_{j-1}, auquel cas, la forme indéterminée obtenue $\frac{0}{0}$ est fixée à 1. On considère dans ce cas que le module M_j ne peut être responsable dans la mesure où il ne produit aucune dégradation locale des performances.

Dans le cas où l'indice de responsabilité est égal à 1 pour l'ensemble des modules, aucun diagnostic n'est effectué pour cette caractéristique. Celle-ci n'est donc pas prise en compte dans le vote final.

2.2.3 Quelques exemples

Pour illustrer cette approche, nous choisissons le module de détection de visage déjà présenté. Les caractéristiques visuelles sélectionnées ici sont alors les suivantes :

1. quantité de bruit,
2. rotation du visage (autour l'axe de la caméra),
3. rotation du visage (autour de l'axe "vertical de l'image"),
4. degré d'occlusion du visage,
5. teinte de la peau (*Hue*),
6. saturation de la peau,
7. contraste,
8. luminosité,
9. résolution.

La méthodologie de sélection des caractéristiques visuelles sera exposée dans la partie suivante. Il y sera montré que certaines des caractéristiques utilisées ici ne peuvent l'être lors d'une application "intégrale" de la méthodologie (c'est à dire en phase avec une analyse des classes de comportement) pour des raisons pratiques relatives à la capacité des caractéristiques à être mesurées sur des images réelles (citons ici le cas problématique du degré de rotation du visage ou encore de la quantité de bruit). Pour autant, ces caractéristiques seront conservées ici à titre explicatif.
Le module de détection de visage utilisé se compose de 4 modules de niveau inférieur (cf chapitre 1 de la partie 2, partie 1.1). Les courbes de la figure 2.7 illustrent les variations de la qualité des résultats en sortie des modules 1 et 2 pour différentes caractéristiques.

Pour illustrer le calcul de l'indice de responsabilité, deux caractéristiques particulières sont considérées : la résolution ainsi que la teinte de l'objet. Les figures 2.8 et 2.9 montrent alors l'évolution des résultats des quatre modules pour ces deux caractéristiques.

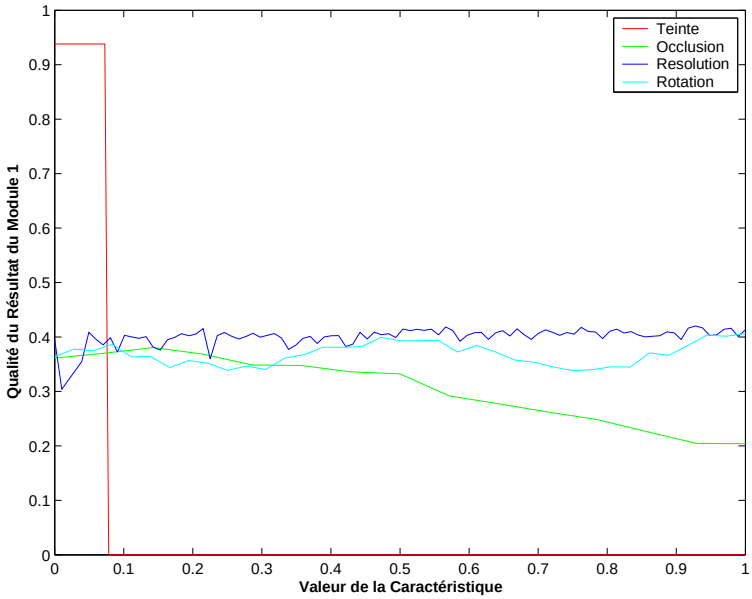
Le tableau 2.3 montre les différents indices de responsabilité obtenus pour les quatres modules dans les plages de chacune des deux caractéristiques considérées.

	M0	M1	M2	M3
Résolution [0.002,0.004]	1	**-4.5507**	0.4151	1
Teinte [0.4,0.5]	$\mathbf{2.4 \times 10^{-16}}$	1	1	1

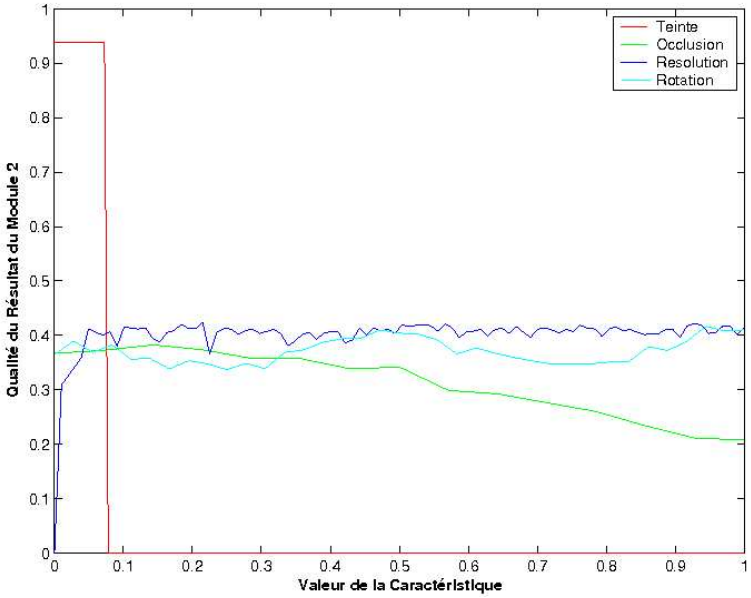
TAB. 2.3 – Indices de responsabilité obtenus pour les différents modules au regard de deux caractéristiques : la résolution et la teinte des visages

Les plages considérées ici sont inclues dans les plages de rejet des filtres associés à ces caractéristiques. Or le fonctionnement des différents modules est connu. Il apparaît ainsi que dans les deux cas, l'indice de responsabilité tend à cibler le bon module (le module M1 pratique un filtre géométrique et dépend de la résolution, le module M0 pratique quant à lui un filtre sur la teinte et la saturation des visages dans l'espace HSL et dépend donc de la teinte des visages.).

Nous venons d'illustrer la première phase de diagnostic, lorsque chaque caractéristique est considérée séparément. Concernant l'étape de fusion des différents diagnostics, la figure 2.10 montre les résultats du vote en fonction des plages de variation des caractéristiques *résolution* et *teinte*. Il est à remarquer que sur la plage [0,0.08] de la teinte, tous les indices de responsabilité attribués aux différents modules sont égaux à 1. En conséquence, seul le diagnostic établi selon la caractéristique résolution est pris en compte dans le vote final et ce dernier attribue la responsabilité au module M1. Dans cette figure, la zone "fonctionnement correct" correspond aux plages de la teinte et de la résolution sur lesquelles le système obtient des résultats corrects et donc, sur lesquelles aucune adaptation n'est nécessaire.

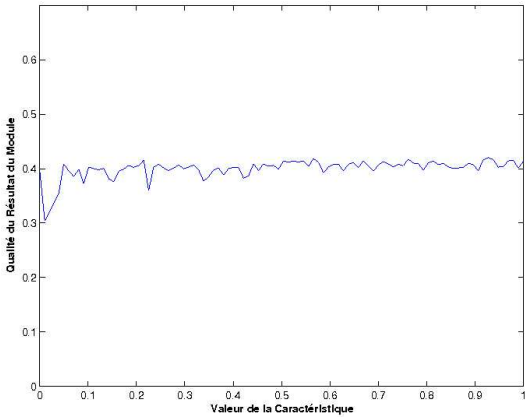


(a) Résultats du module M0 de détection de visages

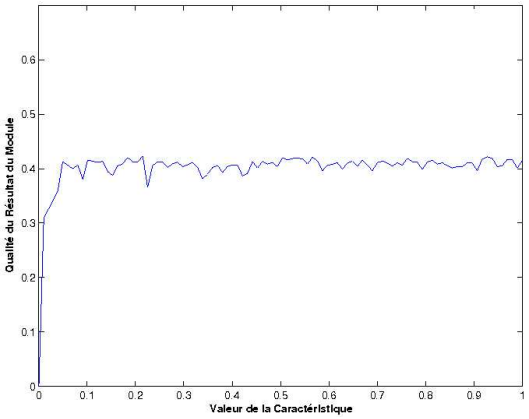


(b) Résultats du module M1 de détection de visages

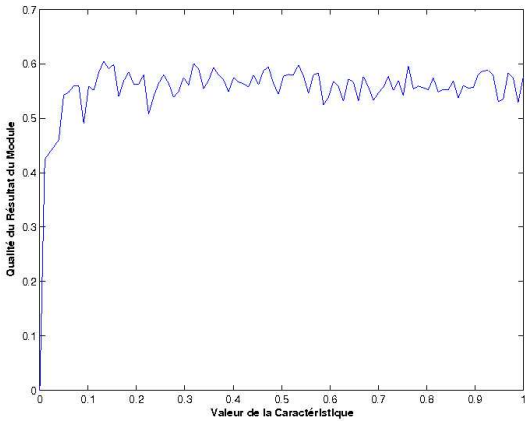
FIG. 2.7 – Evolution de la qualité des résultats de deux modules de niveau inférieur composant la séquence du module de détection de visage pour différentes caractéristiques : on constate notamment que le module M1 (qui filtre les composantes connexes) présente une sensibilité à la résolution des visages (il filtre les zones de petite taille), ce qui n'est pas le cas du module M0



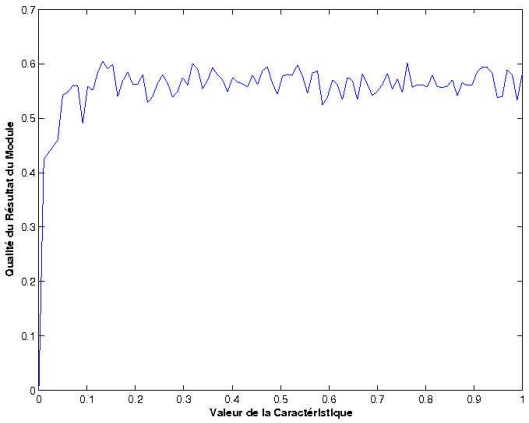
(a) Résultats du Module M0



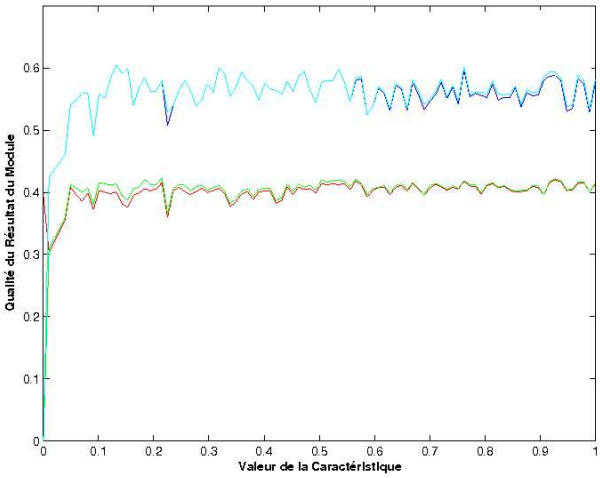
(b) Résultats du Module M1



(c) Résultats du Module M2



(d) Résultats du Module M3



(e) Résultats de l'ensemble des modules (M0 à M3)

FIG. 2.8 – Résultats obtenus par les quatre modules de détection de visages selon différentes valeurs de la caractéristique *Résolution*

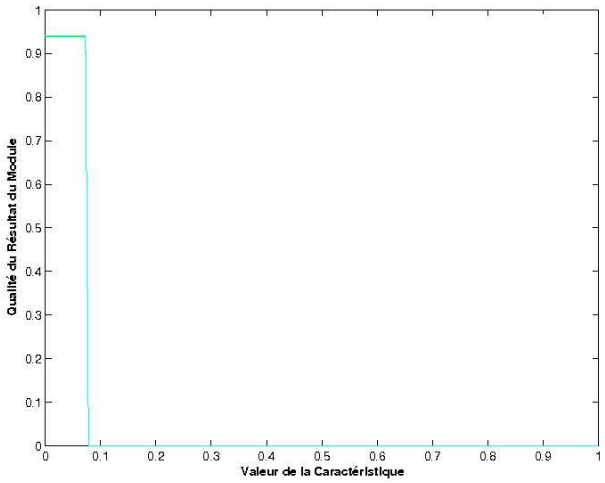


FIG. 2.9 – Résultats obtenus par les quatre modules de détection de visages selon différentes valeurs de la caractéristique *Teinte* (les courbes sont très proches et il est difficile de les différencier : les résultats évoluent très peu à l'issue du module M0)

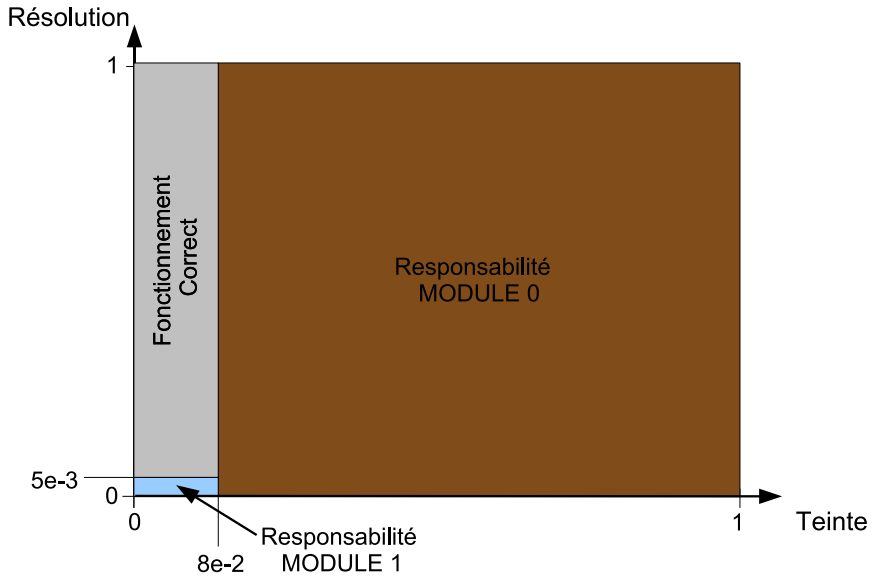


FIG. 2.10 – diagnostic de responsabilité en fonction des plages de variation des caractéristiques *résolution* et *teinte*

2.2.4 Conclusion

La méthode adoptée pour dresser un diagnostic de responsabilité en vue de limiter la phase d'optimisation *pure*, repose sur des postulats simples et bénéficie par ailleurs d'une mise en oeuvre elle aussi relativement aisée, à la différence de la méthodologie d'étude de la sensibilité.
Par ailleurs, si un unique module responsable est choisi ici, il n'en demeure pas moins qu'il est possible de prendre en compte l'ensemble des modules responsables déterminés avant le diagnostic final pour une optimisation séquentielle de ceux-ci (c'est à dire dans l'ordre de leur mobilisation

dans la séquence), jusqu'à obtention d'un résultat satisfaisant.
Il convient aussi de remarquer qu'une troisième voie, explorée en marge des solutions présentées dans cette partie, consiste à établir le diagnostic de responsabilité en s'appuyant sur des mesures d'évaluation adéquates des résultats finaux d'un module, l'analyse de ces mesures permettant justement de déterminer le module responsable en cas d'erreur. Cette méthodologie, présentée dans l'annexe A et concernant un système de détection et de reconnaissance de texte par un OCR commercial, est efficace lorsque les modules sont peu nombreux et lorsque leur comportement est parfaitement maîtrisé (c'est à dire lorsque le diagnostic peut être établi en se basant uniquement sur les résultats finaux). Lorsque ces conditions ne sont pas remplies, la construction de la mesure sous-jacente à l'établissement du diagnostic devient rapidement très difficile.

Il a été vu que la méthodologie adoptée impose de choisir un ensemble de caractéristiques visuelles auxquelles sont attachées des bases dédiées. La prochaine partie sera ainsi l'occasion de développer les réflexions portées à cette problématique, c'est à dire sur le choix des caractéristiques visuelles et sur la construction des bases leur étant dédiées.

2.3 Le choix des caractéristiques visuelles et la construction des bases de test

2.3.1 Choix des caractéristiques visuelles

En premier lieu, les caractéristiques sont utilisées pour représenter les objets contenus dans les différents classes d'objets correspondant aux comportements insuffisants. La solution la plus simple consiste alors à considérer un large ensemble de caractéristiques duquel sont extraites les caractéristiques les plus à même de représenter les objets. De nombreuses méthodes permettent de pratiquer une telle sélection. Pour autant, ces dernières prennent généralement en compte la topologie de l'espace des classes considéré, c'est à dire que les caractéristiques sont choisies relativement à leur capacité à maximiser la distance inter-classe (soit ici entre les comportements). Or les classes manipulées ont été obtenues selon des critères autres que visuels. La différence entre ces classes est assurée du point de vue des performances du module et non de celui des représentations visuelles des objets qui les composent : une sélection des caractéristiques en vue de maximiser la distance entre les classes n'est donc pas la bonne solution.
Certaines méthodes de type ACP produisent une nouvelle représentation des objets contenus dans les classes indépendante de la topologie de l'espace de ces dernières. Pour autant, ces méthodes reposent sur une projection des données dans un espace dont la nature des axes, c'est à dire des caractéristiques, ne revêt plus nécessairement de réalité "visuelle" compréhensible. Par exemple, l'espace initial des caractéristiques peut être composé de la largeur des objets ainsi que de leur densité de contours verticaux. L'espace de projection produit par une méthode telle que l'ACP correspond alors à une combinaison linéaire des deux caractéristiques précitées ; combinaison n'ayant plus aucune signification "*intelligible*". Or ce trait des caractéristiques est nécessaire pour permettre la construction des bases de test.

En effet, le choix des caractéristiques visuelles est guidé par les impératifs relatifs à la méthodologie de diagnostic de responsabilité. Cette méthodologie nécessite de construire, *manuellement*, des bases d'images contenant des objets dont une unique caractéristique varie. Au final, les caractéristiques choisies se doivent donc d'être :

- **synthétisables**, dans le sens où il est possible de simuler facilement une variation de

celles-ci lors de la construction des bases (sans pour autant faire varier dans le même temps les autres caractéristiques) ; ce qui n'est donc pas envisageable lorsque les caractéristiques correspondent, comme dans le cas de l'ACP à des combinaisons de plusieurs autres caractéristiques,

- **mesurables** dans le sens où la valeur de la caractéristique doit pouvoir être mesurée sur des images réelles (pour pouvoir déterminer la représentation des comportements).

Pour autant, réunir ces deux conditions n'est pas simple comme le suggèrent les deux exemples suivants :

- **Une caractéristique synthétisable mais difficilement mesurable** : il est facile de simuler un bruit gaussien dans une image en convoluant cette dernière avec un masque gaussien. Cependant, le problème inverse, c'est à dire la détermination des caractéristiques de ce masque s'avère beaucoup plus complexe[15].
- **Une caractéristique mesurable mais non synthétisable** : il est aisé de calculer la réponse fréquentielle à un filtre de Fourier mais la création "'manuelle" d'images dont on maîtrise la variation de cette réponse est particulièrement complexe.

Les deux contraintes que nous venons de définir réduisent les possiblités quant à la sélection des caractéristiques. Le choix est finalement orienté par une connaissance des objets manipulés, en prenant en compte des caractéristiques connues auquelles sont dépendants les systèmes (par exemple l'éclairage ou la résolution pour les visages).

2.3.2 Construction des bases dédiées

Soit $\mathcal{C}^{visu} = \{C_i^{visu}\}_{i=1..N^{visu}}$ l'ensemble des caractéristiques visuelles utilisées. Il convient alors de construire pour chacune de ces caractéristiques, une base dédiée $DB_{C_i^{visu}}$ composée d'objets présentant des variations selon l'unique caractéristique C_i^{visu}.

Idéalement, les objets doivent être choisis parmi les éléments de la vérité terrain du corpus d'étude. Néanmoins, dans les cas réels, les différences entres les objets ne se limitent pas à une unique caractéristique. Contrôler pour chaque base $DB_{C_i^{visu}}$, les variations des caractéristiques autres que C_i^{visu} implique de définir des seuils en deça desquels les variations tolérées de celles-ci n'ont pas d'incidence sur la variabilité globale des résultats sur la base.

Pour éviter de devoir prendre de tels *a priori*, les bases doivent être construites de façon **artificielle**. Ce mode de construction se révèle par exemple indispensable pour les visages dans le cadre de l'établissement d'une base relative à la couleur. En effet, des visages *réels* de teintes différentes ne peuvent être de formes strictement identiques. De la même façon, s'il est possible de construire une base dans laquelle la variation de la teinte est contrôlée, il est très difficile de contrôler dans le même temps celle de la saturation. La solution consiste alors à construire la base manuellement, en s'appuyant par exemple sur des ellipses dont la couleur est maîtrisée.

2.4 Conclusion

Nous avons présenté dans cette partie deux méthodologies différentes tendant à établir, pour un comportement insuffisant donné, la nature du module responsable, cible privilégiée de l'adaptation. Une première méthode basée sur l'étude de la sensibilité a montré des limitations trop importantes pour être finalement retenue. Bien qu'elle induise une modélisation complexe

[15] Il existe malgré tout des méthodes lorsque le bruit est supposé constant spatialement, c'est à dire lorsque la variance σ est indépendante de la position dans l'image [MJR90]

des opérateurs, cette méthode s'avère être la plus originale et la plus prometteuse. Il sera ainsi envisagé à l'avenir de résoudre les points qui nous ont empêchés de la mettre en oeuvre.
La méthode adoptée finalement est basée sur l'analyse comparative des courbes de variations des différents modules de niveau inférieur, relativement à un ensemble de caractéristiques visuelles. Un essai de cette méthodologie, effectué sur le système de détection de visages déjà utilisé dans les parties précédentes valide cette approche.
Néanmoins, l'établissement du diagnostic de responsabilité repose sur le choix de caractéristiques visuelles auxquelles sont associées des bases dédiées composées d'objets pour lesquels l'unique caractéristique considérée varie. C'est le choix de ces caractéristiques ainsi que le mode de construction des bases qui constitue le seul "*point noir*" de cette méthode, notamment parce qu'elle impose le choix de caractéristiques **mesurables** et **synthétisables**, limitant ainsi l'ensemble des caractéristiques qu'il est possible de prendre en compte. Cette limitation peut se révéler incommode dans certaines situations et nous tâcherons d'y apporter quelques éléments de solution dans les perspectives de cette étude. Pour autant, il est important de rappeler ici l'importance de la construction de telles bases, indépendamment du contexte de leur utilisation dans notre méthodologie. En effet, l'effort actuel porté sur les tâches d'évaluation grande échelle des systèmes de vision (TrecVideo, FERET, TECHNOVISION, ...) montre tout l'utilité de la construction de corpus de test variés et dont toutes les caractéristiques sont maîtrisées.
Ce chapitre clôt par ailleurs la partie consacrée à la présentation de la méthodologie de préparation à l'optimisation, constituant le coeur de cette étude. La partie suivante aura alors pour enjeu de présenter l'instanciation de cette méthodologie à un autre objet, le *texte vidéo*. Ce choix nous imposera ainsi de présenter dans un premier chapitre les spécificités sémantiques autant que physiques de cet objet, les premières validant l'intérêt particulier qui lui est porté. Enfin, le second chapitre présentera les différents résultats obtenus en tenant compte des spécificités précitées.

Troisième partie

Le cas particulier du texte

1

Particularités sémantiques et physiques de l'objet "*texte*", spécificités de l'instanciation de la méthodologie à cet objet

Sommaire

Si la méthodologie d'adaptation a été illustrée sur l'objet visage, le véritable objet *fil rouge* de sa conception a été le texte, dont les spécificités sémantiques en font une cible privilégiée des systèmes d'extraction d'objets lorsque ceux-ci ont pour objectif d'aider à la documentation des flux audiovisuels.
L'enjeu de cette partie est alors, d'une part de justifier l'intérêt particulier porté à cet objet en se basant sur ses caractéristiques en tant que vecteur d'information ; d'autre part de présenter certaines de ses spécificités physiques en tant que zone particulière de l'image. Ces dernières réflexions nous amèneront ainsi à exposer les différents choix effectués pour l'instanciation de la méthodologie d'adaptation à cet objet. Les questions de la constitution des vérités terrain, du choix des mesures d'évaluation, des caractéristiques visuelles ou encore de la construction des bases leur étant associées, seront ainsi abordées.

Concernant l'analyse *sémantique* des textes, l'attention sera portée à différents niveaux, abordant ainsi le texte successivement comme un objet particulier, en comparaison aux autres objets extractibles, comme une source textuelle différente de celle la transcription automatique

et enfin comme une source d'informations particulière dans l'optique de l'aide au remplissage des notices descriptives, s'appuyant alors notamment sur la mesure classique du $TFIDF$[16].
L'analyse physique des *textes vidéos* sera ensuite l'occasion de préparer à l'exposé des résultats obtenus en appliquant la méthodologie présentée dans la partie précédente à un module de détection de textes dont les différents *rouages* seront exposés dans le prochain chapitre.

[16] *Term Frequency Inverse Document Frequency*

1.1 Caractéristiques sémantiques du texte

1.1.1 Le texte vidéo comme objet particulier

Parmi tous les objets qu'il est possible d'extraire automatiquement des flux audiovisuels, le texte revêt un intérêt particulier du fait de son évidente proximité avec la modalité textuelle, la plus couramment utilisée pour décrire les documents.

En outre, à la différence des visages par exemple, la phase de reconnaissance repose sur des procédés de reconnaissance de formes validés et souvent performants (les systèmes d'OCR constituent parmi les premiers systèmes de reconnaissances de formes effectifs), lesquels permettent de focaliser la tâche de recherche sur les phases précédentes, nommément, la détection, le suivi et l'amélioration. Dans le cas des visages, si le principe de la reconnaissance demeure essentiellement le même, c'est à dire comparer la forme extraite avec des patterns connus permettant de déterminer l'identité du visage, la variabilité des formes de visages, corrélée à l'impossibilité flagrante de constituer des bases de patterns suffisantes à la reconnaissance de visages quelconques, témoigne des limitations inhérentes à la reconnaissance d'un tel objet : un visage ne peut être reconnu que s'il est connu au préalable.

Par ailleurs, les textes lisibles à l'écran naissent généralement d'une volonté informative qui peut ne pas être le fait des visages. En effet, la différence essentielle entre les visages et les textes est que ces derniers peuvent résulter d'un effet de post-production (on parlera alors de textes artificiels ou de textes incrustés, cf définition 15), gage de la volonté du réalisateur de les utiliser comme vecteur d'information et donc de leur lisibilité. Par ailleurs, il est à remarquer que même les textes de scène (cf définition 14) peuvent bénéficier d'un traitement privilégié de la part du réalisateur afin de les mettre en valeur (valeur de plan, durée d'apprition, ...) et de les utiliser comme support informatif (on dira alors que ces textes sont *mis en scène*).
Au contraire, s'il est certain qu'un visage "*important*" au sens de la compréhension du document, sera nécessairement filmé dans des conditions propices à sa reconnaissance (taille du plan, éclairage, ...), l'inverse n'est certainement pas vérifié dans de nombreuses situations : un visage peut apparaître clairement à l'écran sans que celui-ci n'ait une quelconque importance informative.

Définition 14. *Le **texte de scène** correspond à un texte filmé. Selon qu'il résulte d'une volonté informative du réalisateur ou non, le texte de scène peut être respectivement **mis en scène** ou **anecdotique**.*

Définition 15. *Le **texte artificiel** correspond à un effet de post-production.*

1.1.2 Le texte vidéo et la transcription de la bande sonore

La description des documents audiovisuels est une tâche essentiellement multimédia : les informations issues des modalités visuelles, sonores et textuelles doivent être extraites et intégrées pour proposer une interprétation fiable de la nature des documents. Le texte est alors une source d'informations particulièrement efficace en comparaison de la source visuelle.

A titre d'illustration de l'intérêt qu'il convient de porter au texte, la campagne d'évaluation TrecVideo s'attache à la question de son extraction dans les documents audiovisuels. La source d'information que constituent les textes extraits peut alors être utilisée pour l'évaluation de la tâche relative à la recherche de documents. Des expérimentations menées dans le cadre de

la campagne 2002 sont détaillées dans [Wol03]. Bien que la base de données fournie pour ces expérimentations ne soit guère propice à l'extraction du texte puisqu'elle contient de nombreux documents *anciens* dans lesquels le texte apparaît relativement rarement, les résultats montrent que le texte permet d'obtenir des résultats intéressants dans le cadre de la tâche de recherche dans une base. Malgré cela, ces mêmes résultats montrent aussi que ce sont les textes issus de la transcription de la bande sonore qui facilitent le plus la tâche de recherche évaluée. La question est alors d'estimer l'influence réciproque de ces deux sources d'informations, autorisant ainsi à établir un premier diagnostic relatif à l'intérêt de l'extraction des textes vidéos comparativement à ceux de la transcription.

L'importante valeur ajoutée des textes de la transcription est quasiment systématique, ne serait-ce que du fait de la quantité des informations générées par l'ASR (*Automatic Speech Recognition*), rapportée à celle bien moindre, générée par l'extraction du texte dans les vidéos (cette constatation est d'autant plus vraie pour les contenus utilisés lors de la campagne d'évaluation TrecVideo de 2002). Cependant, les systèmes ASR actuels fonctionnent avec un vocabulaire fermé fixé généralement dans le cas du français à 65 535 mots. La question de la reconnaissance des termes qui n'appartiennent pas initialement à ce vocabulaire se trouve alors levée, le problème le plus *classique* concernant les entités nommées dont le nombre et la nature varient fréquemment selon l'actualité. Des méthodes d'adaptation dynamique des vocabulaires existent pour limiter la contrainte de leur taille fixe [All03]. Néanmoins, de telles méthodes ne permettent pas de s'absoudre totalement de la limitation posée par cette taille. Le texte vidéo apparaît alors comme une source textuelle complémentaire idéale à celle des textes issus de l'ASR. Dans le cas des journaux télévisés en particulier, il est extrêmement courant que les personnes ou les lieux visibles à l'écran soient associés à des textes les décrivant. Les quelques images suivantes (figure 1.1) illustrent ce trait particulier des textes vidéo, chacune des images composant une réponse à une question à laquelle il pourrait être plus difficile de répondre en s'appuyant uniquement sur les résultats de la transcription automatique.

(a) Qui parle ?

(b) Où se situe l'action ?

(c) Quand se situe l'action ?

FIG. 1.1 – L'objet texte comme support descriptif des images

Une expérience très simple permet de quantifier plus précisément l'apport relatif du texte vidéo en comparaison avec les textes issus de la transcription de la bande sonore. Un corpus de 18 journaux télévisés, contenant 18084 mots différents issus de textes artificiels est constitué. Les entités nommées sont par la suite isolées[17] en nombre de 2657 dont quelques exemples sont

[17] Pour éviter une recherche trop laborieuse, un correcteur orthographique commercial a ici été utilisé, les termes pour lesquels une correction est proposée étant considérés comme des noms propres. Cette méthode présente malheureusement de nombreuses limitations (certains prénoms sont par exemple contenus dans le dictionnaire

donnés dans la liste 1.2.

```
lyon
lorient
lille
mans
caen
Rassinoux
Lemesle
Chérèque
cgt
Rocca
Dy
Aroussi
bornand
Al-Qaida
Fillon-Thibaut
Pujadas
```

FIG. 1.2 – Quelques noms propres contenus dans la liste des textes vidéos issus de la vérité terrain

La proportion de 14% d'entités nommées mesurée sur ce corpus constitue ainsi un premier indice de l'intérêt à porter aux textes vidéos en support aux textes issus de l'ASR.
Une étude plus précise relève de la comparaison entre les textes vidéos et les textes présents dans le vocabulaire "classiquement" utilisé dans le cadre de la transcription de la bande sonore. Il apparaît alors que 52% des termes contenus dans la vérité terrain n'apparaissent pas dans le vocabulaire utilisé pour la transcription. Parmi la liste des textes vidéos absents de ce vocabulaire, on notera (outre les erreurs dans le relevé manuel de ces textes), quelques termes particuliers tels que les nombres (329 termes), certains textes de scène mal orthographiés ou occlus [18] (banderole de manifestation par exemple), et enfin un grand nombre de noms propres parmi lesquels on retrouve certains de la liste établie précédemment (certains d'entre eux, parmi les plus courants, figurent déjà dans le vocabulaire utilisé pour la transcription).

Un dernier atout des textes vidéos est qu'il sont couramment alignés avec la modalité visuelle, alignement qui est loin d'être effectif aussi souvent en ce qui concerne les textes issus de l'ASR. Ainsi, s'il est quasi systématique qu'un texte vidéo désignant le nom d'une personne apparaisse à l'écran en deça de la dite personne, la prononciation de ce nom dans la bande sonore intervient plus rarement durant un segment temporel au cours duquel le visage est visible.

En conclusion, la valeur des textes vidéo a été établie relativement à celle des textes issus d'un système ASR. Pour étayer cette première validation de leur intérêt, il nous reste désormais à évaluer leur apport dans le cadre de l'aide au remplissage des notices descriptives créées à l'Institut National de l'Audiovisuel.

et certains noms peuvent correspondre à des noms communs (Monsieur Chaise)). L'utilisation de méthodes plus robustes constitue ainsi une perspective à cette analyse [BSW99, ZS02].

[18] les textes sont relevés tels qu'ils apparaissent à l'image, en prenant en compte les éventuelles erreurs ou occlusions : ils correspondent ainsi aux résultats attendus (vérité terrain) d'un système DSRO dédié au texte.

1.1.3 Le texte vidéo et l'aide au remplissage des notices

Comme cela a déjà été mentionné (dans le chapitre 1 de la première partie), la documentation s'appuie à l'Institut National de l'Audiovisuel, sur des notices descriptives qui se substituent au document audiovisuel lors, notamment, de la recherche des documents dans la base de données. Aider à la constitution de ces notices est donc un objectif essentiel dans le contexte plus général de l'aide à la documentation. Il est alors évident que le texte vidéo est une source d'informations intéressante dans ce contexte. Pour autant, tous les textes vidéo ne véhiculent pas des informations à forte valeur descriptive (on pensera par exemple aux textes de scènes **anecdotiques** : inscriptions sur les devantures de magasins, textes sur les panneaux de signalisation, ...). Il est ainsi nécessaire d'établir une distinction entre les textes vidéo ayant une valeur descriptive forte et ceux qui, au contraire, ne véhiculent pas des informations suffisamment intéressantes pour être retenus lors de la description du document. Cette distinction nous permettra donc de donner un indice (notamment en termes quantitatifs) de l'utilité des textes vidéo relativement à la tâche de description des documents.
Deux approches ont été envisagées pour évaluer la valeur descriptive des textes vidéos d'un document : la première s'appuie sur le calcul du score TFIDF de ces derniers relativement à une base constituée de notices ; la seconde, dont les premiers développements sont présentés ci-dessous, est empirique et relève de la constitution d'une typologie des textes vidéos.

1.1.3.1 Valeur descriptive des textes vidéos et scores TFIDF

L'ensemble des textes vidéos pourraient être utilisés dans le cadre de l'aide au remplissage des notices. Pour autant, tous ces textes n'ont pas la même valeur descriptive. Partant du principe qu'il est difficile de déterminer *a priori* quels sont ceux qu'il pourrait être intéressant de conserver dans une notice, l'expérimentation menée ici s'appuie sur un corpus composé de documents pour lesquels les notices ont déjà été rédigées.
Un document particulier est alors considéré. Les textes vidéos en sont extraits manuellement. Sont conservés par la suite uniquement ceux apparaissant dans le même temps dans la notice descriptive de ce document. Le résultat de ce filtrage montre alors la proportion des textes vidéos pouvant être utilisés dans le cadre de l'aide au remplissage de la notice considérée.
Par la suite, il est nécessaire d'aller plus avant dans la qualification des textes conservés en recherchant ceux ayant une forte valeur descriptive. Cette valeur dépend alors de la capacité du texte à discriminer le document au cours duquel il apparaît dans une large base (un tel texte peut être par exemple un nom propre peu courant). Cette discrimination relève alors des notices puisque la recherche s'effectue par le biais de ces descriptions : un texte possède ainsi une forte valeur descriptive s'il permet de retrouver la notice du document auquel il appartient dans une base de notices importante. La mesure de cette capacité est alors obtenue par le calcul de l'indice TFIDF de ce texte, indice dont la formule est détaillée dans l'annexe A. On en retiendra ici uniquement qu'il prend en compte le nombre de notices dans lequel un texte apparaît et sa fréquence d'apparition dans la notice considérée : plus le premier est faible et le second important, plus le score TFIDF du texte est élevé et le texte discriminant.

Le protocole d'expérimentation est alors le suivant : une base XML de 58093 notices est créée. Un document particulier de ce corpus est choisi pour l'expérimentation. Ce dernier contient 595 termes différents. 286 d'entre eux apparaissent dans le même temps dans la notice de ce document, soit 48% du nombre total de textes, donnant ainsi un premier indice de la valeur descriptive des textes vidéo. Ces textes filtrés sont ensuite utilisés comme requête sur la base de notices. Les

résultats déterminent pour chaque texte un score TFIDF. La distribution de ces scores apparaît dans la figure 1.3 sous la forme d'un histogramme normalisé, autant du point de vue des scores TFIDF (entre 0 et 1) que de celui des ordonnées, établies en termes de proportion du nombre total de termes.

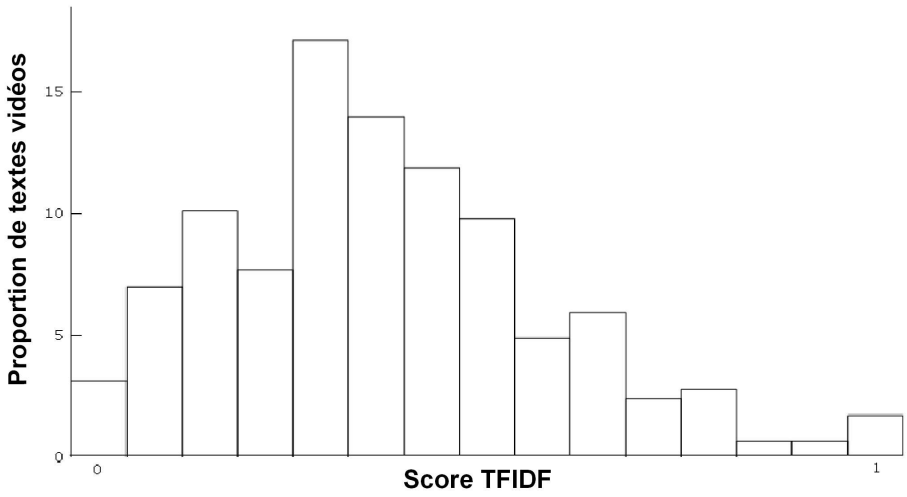


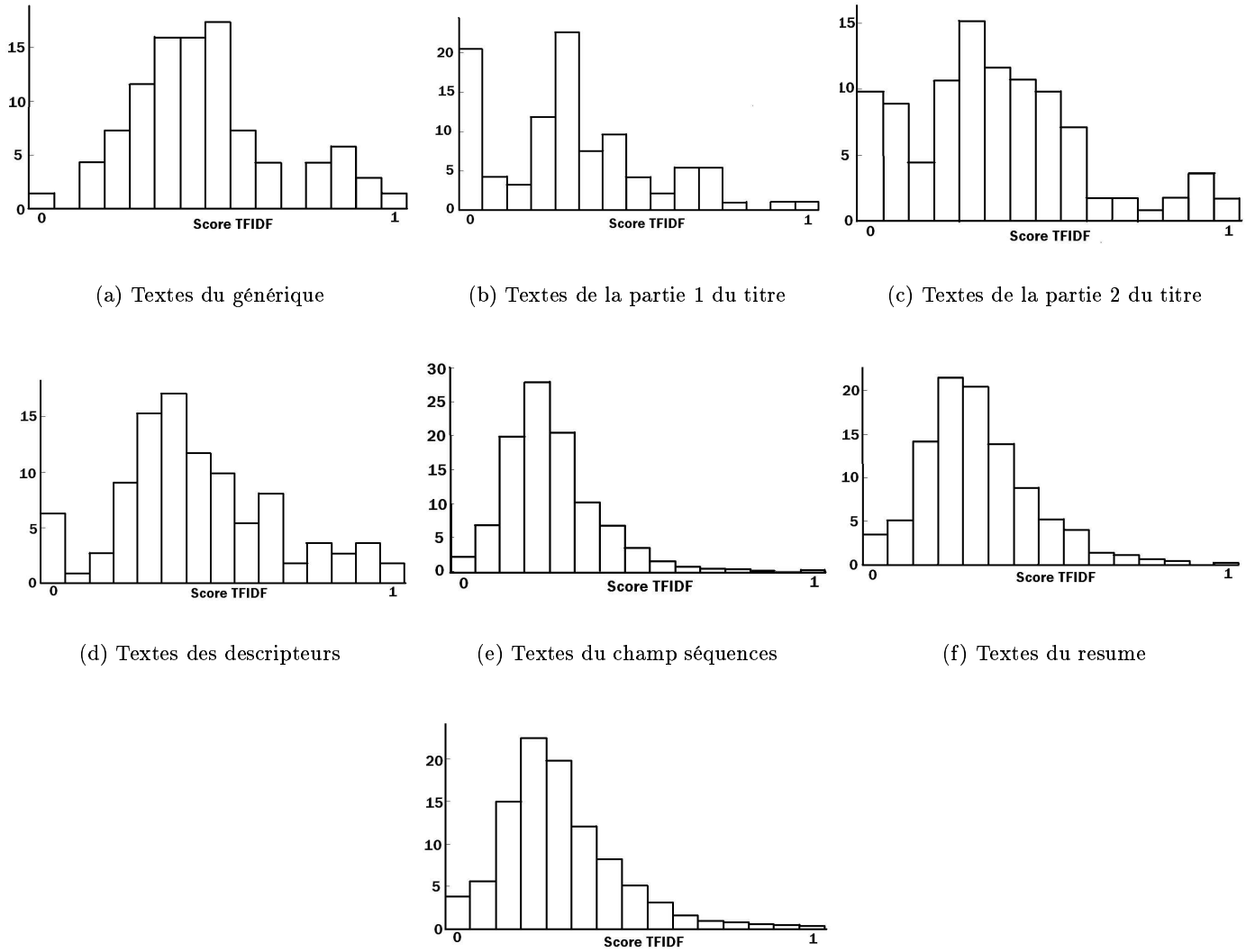
FIG. 1.3 – Répartition des scores TFIDF obtenus pour l'ensemble des termes d'un document de la base

La solution la plus simple, pour établir la proportion de textes à forte valeur descriptive, consiste à seuiller cet histogramme. Pour éviter de devoir choisir un seuil (tâche subjective s'il en est), une solution alternative est préférée, basée sur la comparaison de cette distribution avec celles des scores TFIDF des termes contenus dans la notice descriptive de ce document. Il est en effet supposé ici que la description effectuée par l'entremise de cette notice constitue en quelque sorte la vérité terrain de la meilleure description possible du document[19].
Une notice descriptive est divisée en différents champs. Le mode de rédaction de ces derniers n'étant pas le même (le "'*résumé*"' est par exemple un texte "'*construit*"' à la différence du champ "'*descripteurs*"' composé des mots isolés), la distribution des scores TFIDF des termes qu'ils contiennent peut différer et la distribution montrée dans la figure 1.3 doit donc être comparée séparément avec chacune de ces différentes distributions. Sont considérés ici les champs : *descripteurs*, *résumé*, *séquences*, *générique* et *titre* (globalement, les champs relatifs à la production et au matériel sont ignorés).
Le tableau 1.1 présente le nombre de mots dans chacune de ces catégories tandis que la figure 1.4 montre les différents histogrammes normalisés obtenus relativement aux scores TFIDF de ces mêmes catégories (ces scores sont toujours évalués sur la même base de documents).

Générique	**Partie 1 du titre**	**Partie 2 du titre**	**Résumé**	**Séquences**	**Descripteurs**
69	93	112	111	1092	1003

TAB. 1.1 – Répartition des termes de la notice en fonction des différentes classes.

[19]à défaut de pouvoir établir *a priori* une description idéale, étant donné que la nature de celle-ci dépend, comme cela a déjà été souligné dans le premier chapitre, de l'usage qui en est effectué.



(a) Textes du générique

(b) Textes de la partie 1 du titre

(c) Textes de la partie 2 du titre

(d) Textes des descripteurs

(e) Textes du champ séquences

(f) Textes du resume

(g) Tous les textes

FIG. 1.4 – Histogrammes normalisés montrant la répartition des scores TFIDF pour les différentes classes de textes.

Chacun de ces histogrammes est ensuite comparé à celui représentant la distribution des scores TFIDF des textes vidéo par un test du χ^2 [Sch97], la statistique utilisée étant :

$$\chi^2_{H^0,H^1}(H^0H^1) = \sum_i \frac{(H_i^0 - H_i^1)^2}{H_i^0 + H_i^1}$$

où i désigne l'index du "*bin*" de l'histogramme. Le degré d'association entre les deux distributions modélisées par les histogrammes H^0 et H^1 est inversement proportionnel à la valeur de $\chi^2_{H^0H^1}(H^0H^1)$. Les mesures du χ^2 obtenues sont alors résumées dans le tableau 1.2.

Au delà de la proportion initiale de textes vidéo exploitables puisqu'apparaissant aussi dans la notice (48%), un nouvel indice de la valeur descriptive des textes vidéos est donné par l'analyse du tableau 1.2, puisque l'une des meilleures associations entre la distribution des scores

	Générique	Titre1	Titre2	Descripteurs	Résumé	Séquences	Global
Textes Vidéos	2274	2672	**1295**	**1430**	1650	3259	1653

TAB. 1.2 – Mesures du χ^2 entre les textes vidéos et les textes issus des notices

TFIDF des textes vidéo et celles des différentes catégories de textes des notices est obtenue pour le champ "Descripteurs" portant des informations discriminantes puisque choisies par les documentalistes pour représenter le contenu du document en son ensemble.

Cette étude motive ainsi l'extraction des textes vidéos en tant que vecteurs d'informations. Nécessitant l'utilisation d'une base de notices dont la manipulation s'avère contraignante, la méthode utilisée ci-dessus repose par ailleurs sur l'*a priori* selon lequel les notices constituent la meilleure description des documents qu'il est possible de produire. Sujettes à la subjectivité des documentalistes, il n'existe aucune assurance que celles-ci remplissent parfaitement tous les rôles qui peuvent leur être assignés, tant leur usage dépend d'un contexte applicatif précis. Une solution à ce problème est présentée dans la suite de cette partie. Elle repose sur une typologie des textes vidéos permettant d'évaluer la qualité des informations véhiculées par ceux-ci.

1.1.3.2 Valeur descriptive des textes vidéos et typologie(s)

L'analyse des textes vidéos proposée ici est purement empirique : une typologie de ces derniers (cf figure 1.5), mêlant concepts et relations à la manière d'une ontologie, a été construite dans l'objectif de dresser une cartographie précise de ces objets ou plus précisément, de décrire leur portée sémantique[20]. L'évaluation de la valeur descriptive des textes vidéos peut alors être établie en assignant à chacune des catégories de cette typologie un poids, image de la volonté de l'utilisateur de voir une catégorie particulière de texte être correctement détectée et reconnue par le système.
Bien qu'essentiellement manuelle, cette méthode permet une plus grande flexibilité dans la détermination de la qualité des informations véhiculées par les textes d'un document particulier. Au delà du cadre précité, il est par ailleurs envisageable de coupler cette typologie à une typologie physique des textes, fusion qui permettrait *in fine* de restreindre l'adaptation aux seuls textes appartenant à la catégorie sémantique choisie dans la typologie[21].

1.2 Caractéristiques physiques des textes vidéos et instanciation de la méthodologie

1.2.1 Quelques particularités physiques du texte

Contrairement à ce que l'on pourrait penser, le texte vidéo présente une variabilité toute comparable à celle d'un objet tel qu'un visage. Si cette variabilité peut être plus réduite pour les textes artificiels (du fait de la contrainte de leur lisibilité), ses limites s'avèrent beaucoup plus ténues en ce qui concerne les textes de scène (mis en scène ou non). La figure 1.6 montre différents textes de scène, choisis pour illustrer la diversité des textes de cette catégorie.

[20]Cette typologie a été établie relativement aux textes vidéos apparaissant dans les journaux télévisés, documents qui contiennent souvent de nombreux textes.

[21]On remarquera ici qu'un concept "est de type physique"' est déjà utilisé dans la typologie. Ce concept est introduit ici uniquement pour différencier les textes n'ayant aucun valeur descriptive.

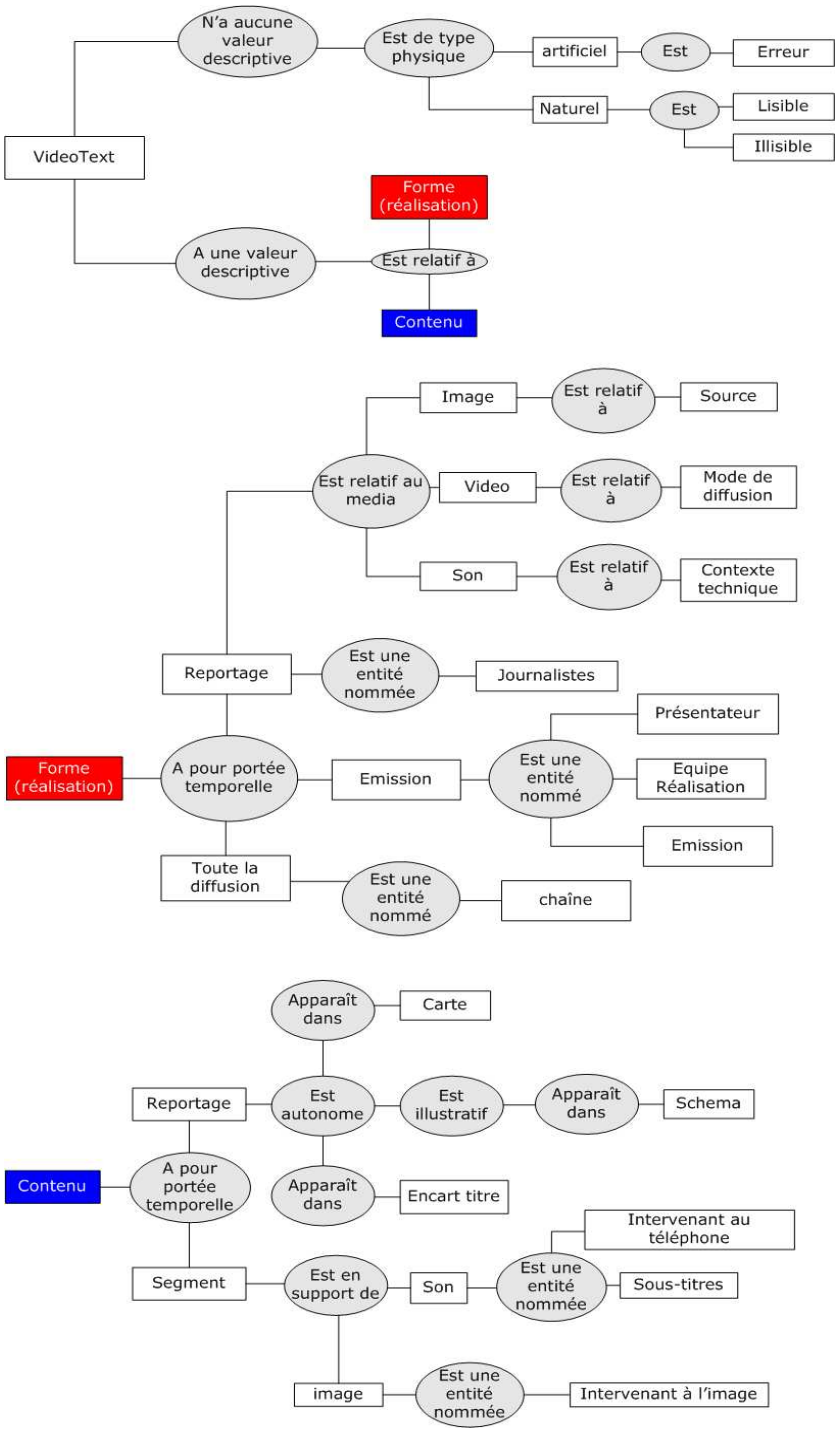


FIG. 1.5 – Arbres présentant la typologie des textes vidéos adoptée

(a) (b) (c)

(d) (e) (f)

FIG. 1.6 – Différents textes de scène

Cette diversité justifie alors *a priori* la nécessité d'appliquer notre méthodologie d'adaptation puisqu'elle illustre parfaitement la difficulté à construire un modèle adapté à la détection de l'ensemble de ces objets. Elle justifie par ailleurs du besoin de pratiquer l'extraction des comportements relativement à des caractéristiques liées aux performances des modules et non à la nature des textes, étant donné la relative difficulté à construire des classes homogènes relatives aux textes de scène.

La spécificité principale du texte vidéo réside dans les différentes granularités auxquelles il peut être appréhendé : à la manière d'un visage dont on peut rechercher à détecter la forme entière ou certains éléments particuliers (généralement les yeux, le nez et la bouche du point de vue des systèmes de détection), il n'existe pas un unique niveau auquel le texte peut être détecté : les lettres, mots, lignes ou blocs de texte constituent autant de granularités auquel cet objet peut être appréhendé. A la différence des visages, le niveau de détection pour lequel opte le système n'est pas systématiquement prédéfini : il arrive ainsi qu'un même système produise des résultats selon différents niveaux, montrant alors généralement une certaine sensibilité à la résolution des textes. Il sera montré par la suite que cette spécificité est exploitée lors de la construction des vérités terrains et lors de la détermination des mesures d'évaluation. Enfin, il est intéressant de remarquer que le texte peut être, à la différence de nombreux objets, considéré comme une

texture particulière dont la fréquence est justement le reflet de ces différentes granularités.

1.2.2 Instanciation

1.2.2.1 Construction des vérités terrain

La construction d'une vérité terrain est, par essence, particulièrement complexe et surtout extrêmement fastidieuse. Des règles précises d'annotation ont donc été établies, et un outil aidant à leur construction a été développé. Du point de vue humain, une unique personne a été en charge de l'annotation. Il eu été sans doute préférable de mettre en place un processus d'alignement de différentes vérités terrains produites par plusieurs annotateurs comme cela peut être le cas par exemple dans le domaine de l'annotation des émotions en audio ([CVRD06]). Pour autant, étant donné la durée d'annotation d'un document, il a été impossible de mener une telle expérimentation.

Les régles d'annotation Construire une vérité terrain du texte revient essentiellement à délimiter les textes apparaissant à l'écran à l'aide de rectangles et à déterminer leurs dates d'apparition et de disparition. La maître principe consiste à ne prendre aucun *a priori* sur le mode de fonctionnement du système ; c'est à dire à constituer la vérité terrain correspondant au comportement idéal attendu. Les cas problématiques, le plus souvent liés à l'aspect "*suivi*" de la vérité terrain, sont les suivants :

- **Les cas d'occlusion** : un texte peut, au cours de son apparition, être caché par un objet, réapparaître puis disparaître de nouveau, etc. Dans ce cas, deux positions peuvent être adoptées : selon la première, "*visuelle*", quelque soit la zone du texte visible à l'écran, celle-ci appartient au même objet texte dans la vérité terrain ; selon la seconde, "*orientée contenu*", un texte est différent dès que son contenu visible est modifié. La première solution est adoptée : la vérité terrain est donc construite telle que notre sens de la vision nous l'impose. Un exemple d'un tel cas (qui concerne bien sûr uniquement les textes de scène) est illustré dans les figures 1.7 et 1.8.

(a) (b) (c)

FIG. 1.7 – Un exemple de gestion des cas d'occlusion : le texte se déplace de la droite vers la gauche. Les trois zones dessinées à l'écran sont associées à un même texte en tant qu'objet spatio-temporel

- **Quand est-ce qu'un texte n'est plus visible ?** La solution la plus simple est de considérer un texte disparu lorsque sa qualité de texte n'est plus distincte visuellement. Malheureusement, il est techniquement impossible d'établir la vérité terrain à vitesse

(a) (b) (c)

FIG. 1.8 – Un autre exemple : le texte se déplace de bas en haut

réelle : le flux est donc observé à une vitesse bien moindre pour éviter les oublis. En observant lentement la vidéo, l'annotateur est donc amené à pratiquer de façon inconsciente une reconnaissance de textes qui ne seraient peut être pas reconnus si l'image seule, extraite du contexte, était montrée à une tierce personne. L'interprétation des dates de fin et de début est alors laissée à la seule interprétation de l'annotateur.

Il a été mentionné dans la partie précédente qu'aucun *a priori* n'était pris sur le niveau de détection effectué par le système de détection de texte. La vérité terrain construite est donc hiérarchique et pour chaque bloc de texte, celle-ci contient aussi les lignes et les mots qu'il contient. Cette structure hiérarchique est systématique : lorsqu'un bloc de texte ne contient qu'un unique mot, les délimitations de la ligne et du bloc sont alors égales à celles du mot considéré (il en est de même pour les délimitations du bloc lorsque celui-ci contient une unique ligne de texte). Par ailleurs, dans le cas des textes de scène, la délimitation en ligne et en mot est parfois trop laborieuse. Dans ce cas, seul le niveau du bloc est retenu et les délimitations du niveau de la ligne et du mot lui sont égales. Ces différentes situations sont illustrées dans la figure 1.9.

L'application d'annotation La nécessité de maîtriser complètement le format des vérités terrain tout comme leur construction nous a amené à développer notre propre application d'annotation dont la figure 1.10 montre une capture d'écran[22].

Cette application est principalement constituée d'un player vidéo (1) (acceptant les formats MPEG1 et MPEG2), d'un ensemble de champs textuels permettant de remplir ou de visionner les informations à inclure dans la vérité terrain et d'une liste (2) dans laquelle apparaissent les VideoText créés (la terminologie VideoText fait référence à l'objet utilisé pour conserver les informations concernant un texte apparu à l'écran). Cette liste peut être triée relativement au contenu des textes, à leur nature (mot, ligne ou bloc) ou à leur type (texte de scène ou artificiel) (3). Le fonctionnement est globalement le suivant : lorsqu'un texte apparaît à l'écran, un bouton commande la création d'un nouveau VideoText (4). Il est alors possible de délimiter sa zone d'apparition dans le player. Deux autres boutons permettent de fixer les dates d'apparition et de disparition des textes (5). Par défaut, il est considéré que la position du texte est la même durant toute son apparition : un VideoText est donc constitué, si aucune modification n'est apportée, de deux objets TextMvt (l'objet TextMvt stocke la position du texte à un instant donné)

[22]Dans la description suivante, les données notées (x) réfèrent aux points marqués dans l'image 1.10.

(a) Bloc constitué de plusieurs lignes, constituées de plusieurs mots

(b) Bloc constitué d'une unique ligne

(c) Bloc constitué d'un unique mot

(d) Un cas difficile : seul le bloc est délimité

FIG. 1.9 – La constitution de la vérité terrain et la délimitation des mots, lignes et blocs

représentant chacun la position de début et de fin du texte. Dans le cas où cette position varie dans le temps, il est possible d'ajouter autant de positions intermédaires que désiré, le nombre de TextMvt contenus dans le VideoText augmentant ainsi de la même façon. Le texte contenu dans la zone délimitée est retranscrit dans une zone de texte permettant de prendre en compte les retours à la ligne le cas échéant (6). Pour chaque VideoText, les images de (7) permettent de visionner les images choisies comme images d'apparition et de disparition du texte considéré, ainsi que les images les précédant ou les suivant respectivement, et ceci dans l'objectif de vérifier les dates attribuées.

La nature du texte ainsi que son type peuvent aussi être spécifiés (si cela n'est pas le cas, un VideoText est par défaut de nature *bloc* et de type *artificiel*). Il est par ailleurs possible d'associer chaque VideoText à un segment particulier du document (typiquement un sujet dans le cas

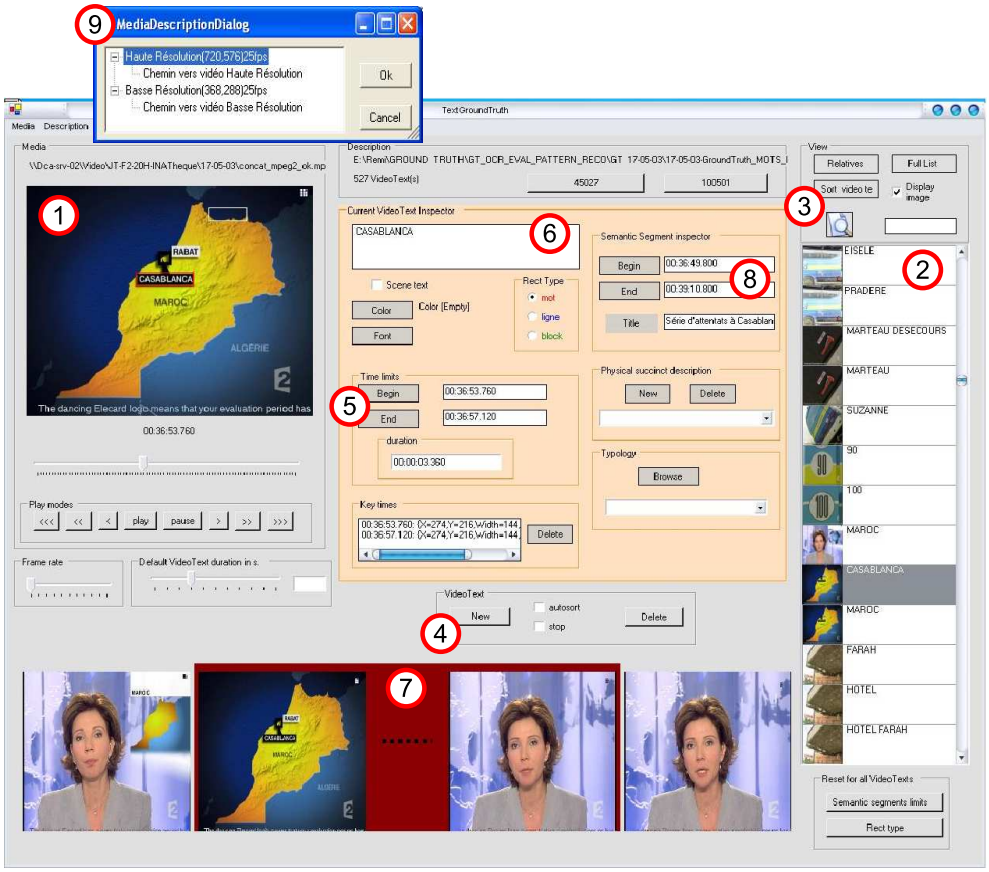


FIG. 1.10 – Capture d'écran de l'application développée pour construire les vérités terrains

des journaux télévisés) (8). La segmentation du document peut être effectuée par l'intermédiaire d'un autre outil qu'il est possible d'appeler depuis l'application d'annotation.

L'ensemble des VideoTexts sont regroupés finalement dans un objet VideoTextDescription qu'il est ensuite possible de sauvegarder au format XML. L'application permet en outre de fusionner plusieurs VideoTextDescription, de les segmenter, etc. Chaque document est potentiellement disponible en plusieurs formats. La VideoTextDescription mentionne l'ensemble des formats disponibles et il est alors possible d'en changer, la taille ainsi que les positions des zones de la vérité terrain étant alors mises automatiquement à jour selon le changement de résolution induit (9).

Enfin, un ensemble de *raccourcis clavier* permet d'accélérer la création des vérités terrain en assignant par exemple à deux VideoText la même position relative au cours de leur apparition ou encore les mêmes dates d'apparition et de disparition.

1.2.2.2 Mesures d'évaluation

La vérité terrain des textes vidéos est constituée de mots, lignes et blocs stockés sous forme hiérarchique. Les mesures d'évaluation proposées dans le chapitre 1 de la partie 2 s'en trouvent modifiées en conséquence : chacune se voit ainsi itérée pour chacun des niveaux considérés. La

mesure d'évaluation d'un texte VT_i, (quelque soit le module qu'elle concerne) devient alors :

$$\mathcal{M}(VT_i) = \begin{pmatrix} \mathcal{M}_{bloc}(VT_i) \\ MOY_{lignes}\left(\mathcal{M}_{ligne}(VT_i)\right) \\ MOY_{mots}\left(\mathcal{M}_{mots}(VT_i)\right) \end{pmatrix} \tag{1.1}$$

Concernant la détection, l'extraction des fausses alarmes et des oublis repose par ailleurs sur les choix suivants :

- Un texte détecté par le module de détection est une fausse alarme s'il ne remplit pas le critère de recouvrement mutuel (cf l'équation 1.2 du chapitre 1 de la partie 2) avec aucun des textes de la vérité terrain de l'image dans laquelle il apparaît, et ceci à aucun des niveaux de la hiérarchie,
- Un texte de la vérité terrain est un oubli si aucun des textes détectés par le module ne remplit le critère de recouvrement mutuel avec une des zones de l'un des niveaux le composant.

1.2.2.3 Le choix des caractéristiques

Le choix des caractéristiques est guidé par les contraintes déjà évoquées, à savoir que les caractéristiques choisies se doivent d'être dans le même temps **mesurables** et **synthetisables**. Les caractéristiques utilisées pour le traitement de l'objet texte sont alors décrites ci-dessous en mettant l'accent sur la question de leur mesure. Les caractéristiques suivantes (elles sont au nombre total de 12) sont calculées sur les blocs de texte, qui constituent le niveau de représentation manipulé dans la méthodologie. Le nom de la caractéristique utilisé par la suite est mentionné entre parenthèses pour chacune d'entre elle.

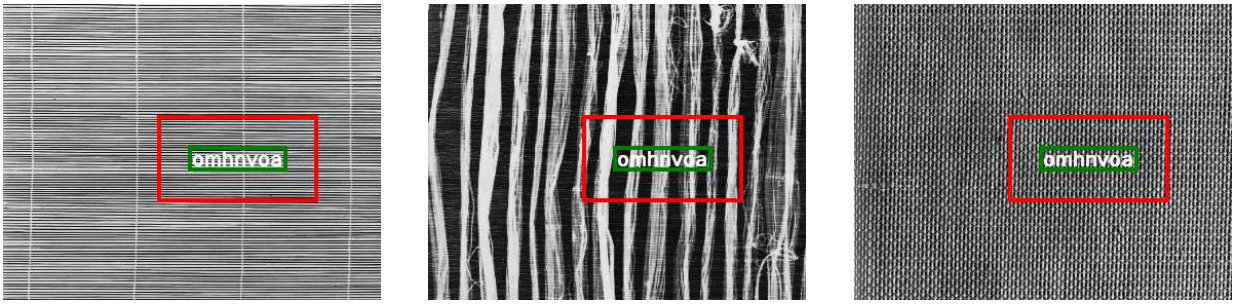
1. **Couleur du texte et du fond** : le module de détection utilisé travaillant en niveau de gris (cf chapitre suivant), ces couleurs correspondent donc à deux niveaux différents. La méthode d'Otsu est utilisée pour déterminer les distributions des niveaux de gris des pixels du texte et du fond, en assimilant chacune de ces deux distributions à une gaussienne. Le niveau du fond correspond alors au niveau de gris moyen de la distribution la plus large (on suppose ici que les pixels de fond sont les plus nombreux) (*couleur1* et *couleur2*).

2. **Contraste** : la méthode de détermination du contraste repose aussi sur l'analyse de la distribution des niveaux de gris. Le contraste est ici assimilé à la différence entre les niveaux les plus distants sur la plage des niveaux de gris et les plus représentés en terme de nombre de pixels. Dans un premier temps, quatre niveaux sont retenus : les deux plus faibles obtenant les *populations* les plus importantes (en terme de nombre de pixels) et les deux plus fortes respectant la même contrainte. Les niveaux de gris moyens de ces deux paires sont calculés et finalement le contraste est défini comme la différence entre ceux-ci, normalisée par le niveau maximal (255 bien entendu) ainsi que par la proportion de pixels mis en jeu pour les quatres niveaux concernés (*contraste*).

3. **Complexité du texte CT et de son fond CF.** Ces mesures sont basées sur celle proposée dans [HWZ04a] : CT = CTH*CTV où CTH correspond à la complexité horizontale et CTV à la complexité verticale du texte avec :

$$CTH = \frac{m_{Sobel_H}}{max_{Sobel_H}} \; et \; CTV = \frac{m_{Sobel_V}}{max_{Sobel_V}}$$

où m_{Sobel} et max_{Sobel} correspondent respectivement à l'intensité moyenne et à l'intensité maximale mesurées dans la zone de texte sur laquelle un opérateur de Sobel vertical ou horizontal a été appliqué. Dans le cas du fond, on différenciera la complexité horizontale de la complexité verticale (CFH et CFV), calculées de façon similaire à la complexité du texte, sur une couronne autour de la zone de texte (la zone de texte agrandie de 30 pixels dans les deux directions). Les images de la figure 1.11 illustrent les mesures obtenues sur des images issues de l'album de Brodatz. La couronne utilisée pour le calcul de CFV et CFH est délimitée dans ces images entre le rectangle vert et le rectangle rouge (CT, CFH et CFV).



(a) CFV= 0.31 et CFH = 0.01 (b) CFV = 0.04 et CFH = 0.17 (c) CFV=0.13 et CFH =0.39

FIG. 1.11 – Mesures de complexité horizontales et verticales sur quelques exemples. Les mesures obtenues montrent que les mesures sont bien à l'image de la densité de lignes verticales (pour la complexité horizontale) et horizontales (pour la complexité verticale)

4. **Orientation des contours du fond** : de la même façon que précédemment, l'orientation du fond est calculée sur une couronne autour de la zone de texte. La détermination de l'orientation repose sur l'application de l'opérateur de Sobel vertical et horizontal sur la zone considérée. L'orientation ($O(p)$) est dans un premier temps calculée en chaque pixel de la couronne :

$$O(p) = atan(\frac{Gy(p)}{Gx(p)})$$

où $Gx(p)$ *et* $Gy(p)$ désignent respectivement la réponse à un opérateur de Sobel horizontal et vertical. Par la suite l'orientation de la zone entière est déterminée en prenant en compte la distribution des angles obtenus : la plage de variation de l'angle est projetée entre 0 et 90 degrés et divisée en segments de 2 degrés. Un histogramme représentant la distribution des angles est produit et le *bin* de cet histogramme comprenant le plus de pixels désigne alors l'orientation de la zone (OC_F).

5. **Position et dimensions de la zone** : ($positionX$, $positionY$, $largeur$, $hauteur$).

6. **Orientation du texte** : seule l'orientation en terme de rotation autour de l'axe de la caméra est prise en compte. La méthode repose sur les travaux présentés dans [MM99] : pour une même zone un ensemble de profils de projection sont calculés pour différentes orientations θ ($\theta \in [-\frac{\Pi}{2}, \frac{\Pi}{2}[$). L'entropie de chacun des histogrammes obtenus est alors

calculée selon la formule suivante :

$$e = -\sum_{i=0}^{N^b} p_i Log_2(p_i) \; avec \; p_i = \frac{N_i}{N}$$

où N correspond au nombre total d'éléments de l'histogramme et N_i au nombre d'éléments du "bin" i de l'histogramme. L'orientation O correspond alors à l'angle pour lequel l'entropie du profil de projection associé est minimale (cf figure 1.12).

$$O = argmin_\theta(e(pp^\theta))$$

où $e(pp^\theta)$ correspond à l'entropie du profil de projection produit pour l'angle θ. La précision de cette mesure dépend alors du nombre d'angles différents calculés (la précision obtenue est ici de $\frac{\pi}{16}$) (0_T).

1.2.2.4 Construction des bases dédiées

Si la construction de certaines bases est immédiate (par exemple celles dédiées à la position, au contraste, etc), il convient de donner quelques précisions pour les bases les plus *problématiques* :

- **Complexité du texte** : utilisation de différentes polices en vue de simuler la variation de la complexité en termes de densité de contours,
- **Complexité du fond** : on différencie la complexité horizontale et verticale en créant des images dont le fond est strié par des lignes (verticales ou horizontales respectivement). Lorsque la fréquence de ces lignes augmente, la complexité associée augmente dans le même temps. Il fut dans un premier temps envisagé d'utiliser des images de textures issues de l'album de Brodatz. Malheureusement, les différences entre les images constituant cette base relèvent de statistiques d'ordre supérieur dont la variation n'est pas contrôlée. La maîtrise "parfaite" des images de la base étant nécessaire, cette solution a été finalement abandonnée,
- **Orientation des contours** : la base se compose d'images dont le fond est strié par des lignes dont l'angle est contrôlé (le nombre en est fixe).

1.2.3 Le choix des caractéristiques et les fausses alarmes

Pour éliminer les fausses alarmes, des filtres relatifs à un ensemble de caractéristiques sont créés (le processus est détaillé dans le chapitre suivant). Le choix de ces caractéristiques n'est pas limité par la contrainte relative à la possibilité de *synthèse* puisque les bases dédiées n'interviennent pas dans la conception de ces filtres. Nous choisissons alors d'utiliser les caractéristiques de bas niveau suivantes :

1. **Caractéristiques colorimétriques** : chaque zone est décomposée en RGB et HSL. Pour chaque canal, la moyenne des valeurs, tout comme les moments d'ordre 2 et 3 sont calculés (18 caractéristiques).
2. **Caractéristiques géométriques** : positions et tailles ainsi que les ratios : $\frac{Y}{X}$ et $\frac{Largeur}{Hauteur}$

Pour optimiser le filtrage, une phase de sélection des caractéristiques, en fonction de leur capacité à distinguer les zones correctement détectées des fausses alarmes, est appliquée. Cette sélection s'appuie sur la méthode de Fisher et produit un classement des 24 caractéristiques en fonction de leur capacité à discriminer les deux classes prises en compte. Pour appliquer cette

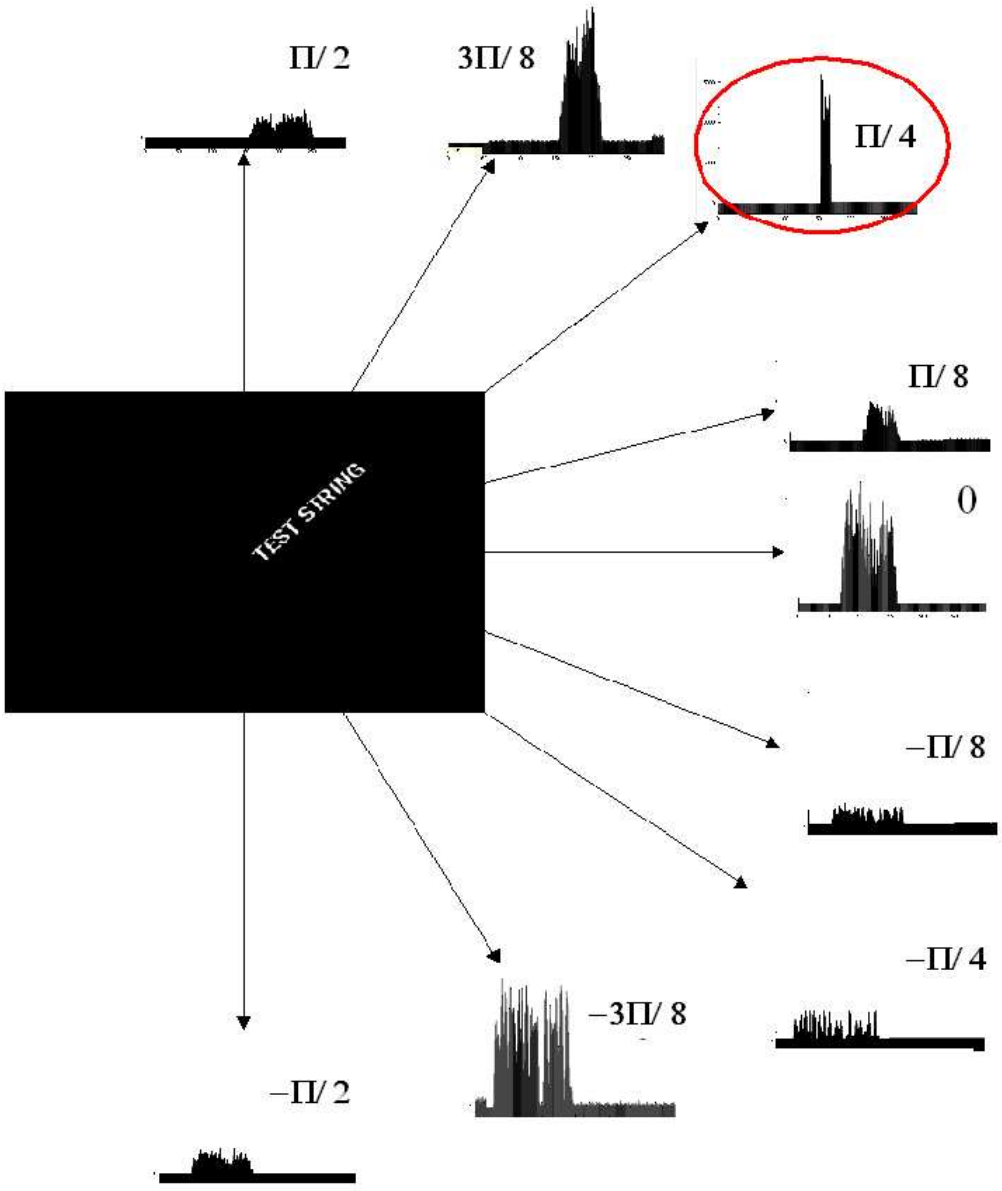


FIG. 1.12 – Estimation de l'orientation du texte à partir de profils de projection

méthode, deux ensembles d'apprentissage comportant chacun 2000 éléments de ces deux classes sont construits. L'utilisation du vecteur obtenu par la méthode de Fisher pour la construction des filtres associés sera détaillée dans le chapitre suivant.

1.3 Conclusion

Nous avons présenté dans cette partie les spécificités sémantiques et physiques du texte ayant conduit respectivement à privilégier cet objet comme *fil rouge* de notre étude et à effectuer les choix concernant l'instanciation de la méthodologie relativement à la construction des vérités terrain, au choix des mesures d'évaluation, des caractéristiques visuelles ainsi qu'à la construction

des bases leur étant dédiées.
Reste désormais à mettre en application la méthodologie au module de détection du texte vidéo, mise en oeuvre qui donne alors lieu à la dernière partie de ce manuscrit dans laquelle les résultats obtenus seront exposés.

2

Résultats des expérimentations sur l'objet "*texte*"

Sommaire

Le principal apport de la méthodologie présentée dans les parties précédentes réside dans la phase de préparation à l'optimisation des paramètres d'un module particulier. Le principe en est le même pour tous les modules d'un système DSRO : que ce soit pour la phase de détection, de suivi, d'amélioration ou encore de reconnaissance. Les expérimentations présentées dans cette partie portent alors sur le module de détection d'un système DSRO consacré à l'objet texte. Subséquement à un état de l'art du domaine de l'extraction des textes vidéo et à la présentation du module utilisé, les résultats obtenus pour les différentes étapes de la méthodologie seront présentés. Ces étapes sont rappelées ci-dessous :

1. **Analyse des comportements** :
   - Constitution des vérités terrains
   - Calcul des mesures d'évaluation
   - Application de l'algorithme de clustering

2. **Analyse du système : diagnostic de responsabilité**
   - Extraction des comportements *insuffisants*
   - Représentation de ces derniers selon les caractéristiques visuelles décrites dans le chapitre précédent
   - Analyse des courbes d'évolution des performances des différents modules de niveau inférieur selon ces caractéristiques pour l'établissement du diagnostic de responsabilité
3. Traitement des fausses alarmes

2.1 Etat de l'art de l'extraction de textes dans les vidéos

L'intérêt de l'extraction des textes vidéo est reconnu et de nombreux systèmes visant à réaliser cette tâche ont été développés. Leur composition en modules est la même que pour tout système DSRO : détection, suivi, amélioration et reconnaissance. Une étape supplémentaire, propre au traitement du texte doit être prise en compte : un module de binarisation préalable à la reconnaissance est ainsi généralement proposé.
Bien que les expérimentations menées par la suite portent uniquement sur la phase de détection, l'ensemble des modules sont abordés ici, permettant ainsi de mieux appréhender les démarches scientifiques, globales, de conception des systèmes. Le tableau 2.1 énumère les modules mis en oeuvre par chacun des systèmes détaillés dans cet état de l'art.

Article	Détection	Suivi	Amélioration	Binarisation	Reconnaissance
[ZKJ95]	X				
[SKH+99]	X	X	X		X
[SBK99]	X			X	
[WMR99]	X			X	
[GA00]	X			X	
[LDK00]	X	X	X	X	
[DADB00]	X	X	X	X	
[XHC+01]	X	X	X	X	
[HCWZ01]	X				
[KJPhK01]	X				
[Jun01]	X				
[LW02]	X	X	X	X	
[TGLZ02]	X	X	X	X	
[WD02]				X	
[WC03]	X			X	
[WK03]	X				
[Wol03]	X	X	X	X	
[COB04]	X			X	

TAB. 2.1 – Nature des modules traités par les systèmes d'extraction de texte issus de la littérature

Il a déjà été expliqué que tout système DSRO repose sur la définition d'un modèle de l'objet cible de l'extraction. C'est ensuite la distance entre une région considérée et le modèle qui permet de prendre une décision quant à l'appartenance de cette région à un objet du type recherché. Les différents systèmes se différencient alors selon les points suivants :

1. Quelles caractéristiques pour le modèle ?

2. Quelles plages de variation pour ces caractéristiques ?
3. Quels opérateurs pour le calcul de ces caractéristiques ?

La troisième question met en jeu des choix technologiques que nous ne détaillerons pas par la suite. Au contraire, la nature des caractéristiques utilisées ainsi que le mode d'instanciation du modèle seront discutés avec précision.

2.1.1 La détection

Les modules de détection s'organisent généralement autour d'une ou plusieurs caractéristiques *centrales* auxquelles peuvent s'ajouter des caractéristiques secondaires permettant d'affiner les résultats lors d'un post-traitement. La nature des *caractéristiques centrales* autour desquelles sont construits les systèmes considérés sont listées dans le tableau 2.2.

Article	*Texture*	*Couleur*	*Géométrie*
[ZKJ95]		X	
[SKH+99]			X
[SBK99]		X	
[WMR99]	X		X
[GA00]	X	X	
[LDK00]	X		
[DADB00]	X	X	
[XHC+01]	X		
[HCWZ01]	X		
[KJPhK01]	X		
[Jun01]	X		
[LW02]	X		
[TGLZ02]		X	
[WC03]	X		X
[WK03]		X	X
[Wol03]	X		X
[COB04]	X		

TAB. 2.2 – *Caractéristiques centrales* des systèmes

Le choix de ces caractéristiques repose sur des heuristiques adoptées par le concepteur :

1. **Texture** : du fait de l'alternance répétée entre caractères et zones "*de fond*", les zones de texte ont une forte densité de contours verticaux ou de coins. L'orientation du gradient ainsi que la fréquence des contours constituent d'autres indices texturaux de la présence de texte.
2. **Couleur**[23] : les zones de texte peuvent être discriminées en fonction de leur contraste, en admettant que ce dernier doit être élevé pour que le texte puisse être lu. En marge de cette caractérisation, deux autres points de vue relatifs à la couleur existent :
   (a) assimiler les zones de texte à des zones de forts changements locaux de couleur en différenciant deux à deux les images contigües du flux considéré ([TGLZ02]),

[23] La couleur est ici comprise en un sens large : certaines méthodes utilisent en fait uniquement l'intensité lumineuse comme caractéristique.

(b) caractériser les zones de texte par l'existence de deux combinaisons de couleurs permettant de produire par quantification des régions satisfaisant par la suite des contraintes géométriques ([ZKJ95] par exemple).

3. **Géométrie** : le texte est composé de lignes, de caractères alignés, d'une ligne de base. Il peut être par ailleurs supposé horizontal. Il est à remarquer que les contraintes concernant la largeur ou encore la hauteur de la zone de texte sont généralement utilisées lors d'une étape de post-traitement et ne sont donc pas considérées comme des caractéristiques centrales.

La construction de la signature des objets textes relativement aux caractéristiques centrales en vue de distinguer ceux-ci de zones quelconques de l'image peut être confiée à un système d'apprentissage. Le tableau 2.3 résume alors les différentes méthodes d'apprentissage utilisées, tout en spécifiant dans le même temps certains des paramètres les plus importantes relatifs à ces dernières.

Article	*Méthode d'apprentissage*
[LDK00]	RN (multi-couches (3), bootstrapping)
[KJPhK01]	SVM (noyau polynomial)
[Jun01]	RN (multi-couches (3), FA sigmoïde, rétropropagation, bootstrapping)
[LW02]	RN (multi-couches (3), FA sigmoïde, bootstrapping)
[TGLZ02]	RN (FCNN[a] multi-couches(3))
[Wol03]	SVM (régression[b], noyau polynomial (degré 3), bootstrapping, validation croisée)
[COB04]	RN (multi-couches (3), FA sigmoïde, rétropropagation, cross-validation) ; SVM (noyau RBF)

TAB. 2.3 – Méthodes d'apprentissage utilisées par les systèmes : *RN* signifie ici Réseau de Neurones, *FA* Fonction d'Activation, *SVM* Support Vectors Machine, *FCNN* Fuzzy Clustering Neural Network

[a] A la différence des autres classifieurs basés sur les réseaux de neurones, chaque zone se voit attribué un degré d'appartenance à chacune des classes considérées.

[b] Réduction de la complexité en approximant l'hyperplan par un hyperplan nécessitant moins de vecteurs supports.

Il est par ailleurs envisageable de confier au mécanisme d'apprentissage un vecteur de caractéristiques constitué uniquement des niveaux de gris de l'image (ou d'une zone particulière) auquel cas la phase de construction des caractéristiques est laissée libre au mécanisme d'apprentissage [KJPhK01, Jun01]. Pour autant, la mise en oeuvre de ces mécanismes impose au concepteur un paramétrage (choix du noyau pour les SVM, fonction d'activation des neurones dans un réseau de neurones) induisant la fonctionnnelle de construction des caractéristiques.

En marge du processus principal organisé autour des caractéristiques centrales, certains systèmes mettent en oeuvre une phase de pré-traitement (cf tableau 2.4) et parfois une analyse multirésolution permettant d'intégrer les résultats obtenus à plusieurs échelles [WMR99, LDK00, XHC$^+$01, KJPhK01, LW02, Wol03].

Finalement, le seul filtrage selon les caractéristiques centrales est souvent insuffisant et une étape de post-traitement est mise en oeuvre pour préciser la délimitation des zones de textes obtenues (par exemple segmentation en lignes ou fusion des zones proches) et supprimer les fausses alarmes (par exemple filtrage selon des critères géométriques). Le tableau 2.5 présente les principaux post-traitements mis en oeuvre par les modules de détection de texte.

Concernant les systèmes mettant en jeu la multirésolution, une phase commune de post-traitement consiste à fusionner les résultats obtenus aux différentes échelles.

Article	*Pré-traitement*
[DADB00]	Accentuation des contours, débruitage
[XHC+01]	Affinage des contours, débruitage
[TGLZ02]	Détection des plans dans le flux vidéo
[WK03]	Normalisation RGB, débruitage

TAB. 2.4 – Pré-traitements mis en oeuvre

Article	*Post-traitement*
[ZKJ95]	filtrage et fusion géométrique, segmentation en lignes
[SKH+99]	lissage, filtrage géométrique
[SBK99]	filtrage géométrique
[WMR99]	filtrage et fusion géométrique
[GA00]	filtrage, fusion et segmentation géométrique
[LDK00]	segmentation en lignes
[DADB00]	filtrage et fusion géométrique, segmentation en lignes
[XHC+01]	ouverture morphologique, segmentation en lignes, filtrage géométrique
[HCWZ01]	dilatation morphologique, filtrage géométrique
[KJPhK01]	lissage, fusion géométrique
[Jun01]	débruitage, filtrage et fusion géométrique.
[LW02]	filtrage et fusion géométrique, segmentation en lignes et mots
[TGLZ02]	segmentation en caractères, lissage
[WC03]	filtrage géométrique, prise en compte des *hampes*, lissage
[WK03]	filtrage et fusion géométrique, filtrage colorimétrique
[Wol03]	affinage morphologique[a], filtrage et fusion géométrique
[COB04]	segmentation en lignes, filtrage géométrique

TAB. 2.5 – Post-Traitements mis en oeuvre

[a]Cette phase est aussi mise en oeuvre dans le système utilisé pour nos expérimentations et sera détaillée dans la partie suivante

2.1.2 Le suivi

Six systèmes proposent un module de suivi des textes dans les vidéos (cf tableau 2.1) :

1. [LW02] : la détection est effectuée toutes les 30 images pour réduire le temps de calcul. Lorsqu'une image donne une réponse positive au détecteur, la détection est appliquée dans les images contigües. Le suivi consiste alors à comparer les zones détectées dans ces différentes images et à les associer selon des critères géométriques. Une signature basée sur les profils de projection valide les associations obtenues.
2. [LDK00] : le suivi est prédictif. Pour chaque zone de texte détectée, la recherche dans l'image suivante est guidée par un modèle translationnel du mouvement des textes. La méthode des moindres carrés permet ensuite d'affiner la position de la zone associée à la zone de texte détectée. Une étape de vérification basée sur les contours valide ensuite l'association. Dans les cas simples de translation, la position du texte dans l'image suivante est prédite beaucoup plus précisément. Cette méthode de suivi est reprise dans [XHC+01].
3. [SKH+99] : les zones détectées dans des images contigües sont associées selon des critères géométriques.

4. [DADB00] : les zones détectées dans des images contigües sont associées selon des critères colorimétriques et géométriques (méthode détaillée dans [SD98]).
5. [TGLZ02] : le texte est supposé immobile. Sa position est déterminée en fonction de celle observée lors de son apparition et de sa disparition.
6. [Wol03] : les zones détectées dans des images contigües sont associées selon des critères géométriques. Toute association est validée par la comparaison de signatures basées sur les profils de projection comme dans [LW02].

L'étape du suivi est par ailleurs souvent l'occasion de supprimer de nouveau des fausses alarmes selon des critères temporels (durée minimale d'apparition d'un texte pour qu'il puisse être lu, nombre maximal de "*trous*" durant le suivi, etc). De tels critères sont par exemple appliqués dans [LW02, Wol03].

2.1.3 L'amélioration

Il est fréquemment reconnu que la principale difficulté à détecter les textes vidéo tient à la complexité du fond sur lequel ceux-ci sont inscrits ainsi qu'à leur faible résolution [LW02, Wol03, SKH+99]. Le module d'amélioration (cf tableau 2.6) s'attache ainsi à réduire la complexité du fond et à augmenter la résolution des textes. Concernant la réduction de la complexité, la méthode la plus courante consiste à intégrer temporellement les instances d'un même texte, en supposant que le fond est plus mobile que le texte.

Article	*Résolution*	*Complexité du fond*
[SKH+99]	Interpolation bi-linéaire	Extrema Temporels
[LDK00]	Interpolation bi-linéaire [LD99]	Moyenne Temporelle [LD99]
[XHC+01]		Moyenne Temporelle
[LW02]	Interpolation	Extrema Temporels
[TGLZ02]	Interpolation par des Splines	Extrema Temporels
[Wol03]	Interpolation bi-cubique	Moyenne Temporelle

TAB. 2.6 – Méthodes d'amélioration mises en oeuvre

On citera enfin, une dernière méthode d'intégration temporelle reposant sur l'opérateur de moyenne, appliquée uniquement sur les zones des images présentant un fort contraste [HYZ02].

2.1.4 La binarisation

Admettant que la plupart des systèmes de reconnaissance obtiennent de meilleurs performances sur des images binaires, un grand soin est apporté par certains systèmes au module de binarisation :

1. [SBK99] : les deux couleurs (fond et texte) sont obtenues par clustering,
2. [WMR99] : l'histogramme de la zone de texte est lissé et la recherche de la vallée la plus marquée permet d'obtenir le seuil de binarisation,
3. [GA00] : 4 couleurs sont extraites par clustering. Une (ou deux) couleur(s) est(sont) attribuée(s) au fond et la combinaison des autres détermine la couleur du texte. Autant d'images binaires qu'il existe de combinaisons sont créées. La sélection du meilleur résultat est effectuée relativement à l'analyse des profils de projection,
4. [LDK00] : seuillage manuel de l'histogramme,

5. [DADB00] : seuillage de l'histogramme. La couleur moyenne du voisinage de la zone de texte (censée contenir des pixels appartenant au fond) détermine le sens du seuillage,
6. [XHC+01] : seuillage adaptatif de Niblack [Nib85] basé sur le contraste. La variation du paramètre k de cette méthode produit plusieurs résultats de binarisation. La sélection se base alors sur les résultats de la reconnaissance,
7. [LW02] : une analyse par "*region growing*" (en prenant comme graine un pixel du bord de la zone supposé appartenir au fond) permet de séparer les pixels de texte des pixels de fond. Une analyse par quantification des couleurs permet ensuite de déterminer si le texte est normal ou inversé (dans le but de choisir le sens de la binarisation comme dans [DADB00]). Un seuil global est ensuite calculé à partir de ces informations,
8. [TGLZ02] : seuillage global de l'histogramme,
9. [WD02] : un estimateur de Bayes de maximum *a posteriori* (par recuit simulé) est utilisé pour produire l'image binaire, en modélisant celle-ci selon des champs de markov aléatoires,
10. [WC03] : comme dans [SBK99], un clustering bi-couleur permet d'obtenir la couleur du fond et celle du texte,
11. [Wol03] : en sus de la méthode développée dans [WD02], une méthode de binarisation basée sur une variation du seuillage adaptatif de Sauvola [SSHP97] (par maximisation du contraste) est proposée,
12. [COB04] : trois méthodes différentes de clustering des couleurs sont expérimentées. Deux de ces méthodes modélisent la distribution de chaque classe de couleur par une gaussienne dont les paramètres sont estimés par un algorithme EM (maximisation de la vraisemblance) ou une variation de l'algorithme EM mettant en jeu une modélisation de la zone par des champs de markov aléatoires. La dernière méthode consiste à appliquer l'algorithme des k-moyennes. Plusieurs nombres de classes sont testés et la meilleure binarisation parmi l'ensemble des résultats obtenus est choisie.

2.1.5 La reconnaissance

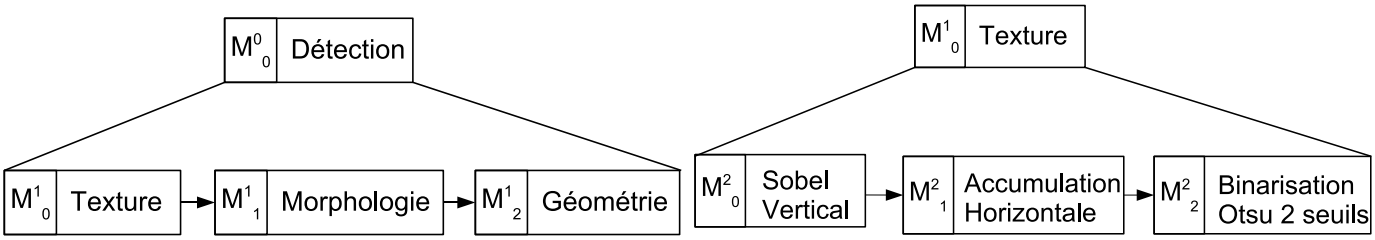
Dans la grande majorité des cas, un système OCR commercial est utilisé lors de la phase de reconnaissance. Certains systèmes mettent en oeuvre une phase de sélection du meilleur résultat de reconnaissance parmi un ensemble produit en faisant varier certains paramètres relatifs à l'extraction des zones de textes.

Dans [COB04] par exemple, plusieurs résultats de reconnaissance sont produits pour les différentes binarisations proposées par clustering des couleurs. Le résultat final est alors choisi en fonction d'un score de confiance prenant en compte deux modèles : le premier s'appuie sur un modèle de langage bi-gramme et détermine la probabilité d'obtenir une chaîne de caractères T avec une bonne segmentation ; le second se base sur un modèle uni-gramme construit en donnant à l'OCR des mauvaises segmentations ou des fausses alarmes et établit la probabilité d'obtenir la chaîne T avec une mauvaise segmentation. Dans les deux cas, un biais est introduit pour tenir compte de la longueur de la chaîne de caractères.
Dans [SKH+99], un système autonome de reconnaissance est développé : chaque zone extraite, représentant un caractère, est ainsi comparée avec des formes de références. Les mesures de similarité avec les formes de référence sont comparées pour plusieurs segmentations différentes des caractères, le résultat de la reconnaissance correspondant alors à la meilleure mesure obtenue. La reconnaissance des mots (et non des caractères) est par ailleurs couplée à l'utilisation de dictionnaires.

2.2 Le module de détection de texte utilisé

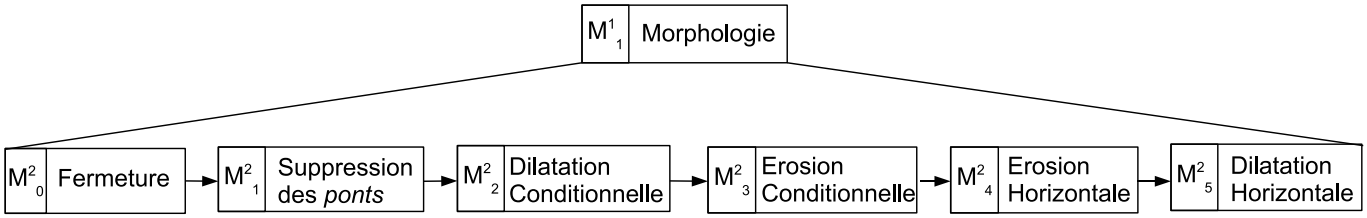
Le module de détection de texte choisi comme support de nos expérimentations est décrit dans [Wol03]. Seule cette phase du système entier d'extraction sera développée ici. Concernant les modules de suivi, d'amélioration et de binarisation, les choix effectués sont les mêmes que pour un second système présenté dans [Wol03] (basé sur l'utilisation des SVM) et ont déjà été précisés dans l'état de l'art.

La figure 2.1 montre la décomposition du module de détection utilisé sur 2 niveaux.

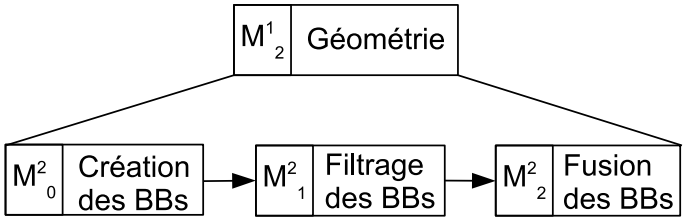


(a) Description du premier niveau de décomposition du module de détection

(b) Décomposition du module lié à la texture



(c) Décomposition du module lié à la morphologie



(d) Décomposition du module lié à la géométrie

FIG. 2.1 – Description du module de détection étudié sur deux niveaux de décomposition. Au niveau 1, les modules sont liés à des caractéristiques particulières du texte : texture, morphologie et géométrie (BB désigne l'appelation anglo-saxonne des boîtes englobantes : "*Bounding Box*").

Voici quelques précisions sur le fonctionnement de ces modules :

1. **Module "*texture*"** : il est supposé ici que les zones de texte présentent une forte réponse à un opérateur de Sobel horizontal, étant composées d'alternances horizontales de gradient (passages caractères/fond, fond/caractère). L'accumulation permet ensuite d'ac-

centuer cette réponse dans les zones de texte. La méthode de binarisation repose sur la méthode d'Otsu assimilant la distribution des niveaux de gris à deux gaussiennes, représentant respectivement les niveaux de gris des pixels du fond et ceux du texte. Pour résoudre toute ambiguïté dans la potentielle zone d'intersection entre les deux gaussiennes, un second seuil est ajouté et un seuillage par hystéresis est effectué.

2. **Module morphologique** :
   (a) La fermeture permet de supprimer les petits "*trous*" potentiellement obtenus à l'issue du module lié à la texture.
   (b) La dilatation conditionnelle a pour objectif de connecter les caractères pour former des mots. Le critère de la dilatation d'un pixel repose donc sur la différence de position et de taille des composantes connexes entourant le pixel courant : si elles sont suffisamment proches et alignées horizontalement, la dilatation est appliquée.
   (c) L'érosion conditionnelle consiste à appliquer l'érosion sur les seuls pixels dilatés lors de la dilatation conditionnelle.
   (d) L'érosion horizontale a pour but de filtrer les composantes selon leur longueur.
   (e) La dilatation horizontale permet de faire recouvrer aux composantes restantes leur taille originelle (avant érosion horizontale).
3. **Module géométrique** :
   Les boîtes englobantes des composantes connexes sont créées et filtrées selon des contraintes géométriques (taille, ratio $\frac{largeur}{hauteur}$). Finalement, les boîtes s'intersectant sont fusionnées selon des critères de surface de recouvrement.

Le tableau 2.7 dénombre alors les paramètres utilisés par les différents modules de niveau 2.

Module de Niveau 1	**Module de niveau 2**	**Paramètres**
Texture	Sobel vertical	0
	Accumulation	1 (S)
	Binarisation	1 (α)
TOTAL	2	
Morphologie	Fermeture	1 (N_F)
	Suppression des ponts	2 (N_{SP}, Th_1)
	Dilatation conditionnelle	3 ((N_{DC}, Th_2,Th_3)
	Erosion conditionnelle	1 (N_{EC})
	Erosion horizontale	1 (N_{EH})
	Dilatation horizontale	1 (N_{DH})
TOTAL	9	
Géométrie	Création des boîtes	2 (δ_X,δ_Y)
	Filtrage	2 (Th_4,Th_5)
	Fusion	3 (Th_6,Th_7,Th_8)
TOTAL	7	
TOTAL DES TOTAUX	**18**	

TAB. 2.7 – Paramètres des différents modules de niveau 2

- S correspond à la taille de la fenêtre d'accumulation,
- N_F, N_{SP}, N_{DC}, N_{EC}, N_{EH} et N_{DH} désignent le nombre d'itérations de l'opérateur morphologique auquel ils sont attachés,

- Th_1, Th_2, Th_3, Th_4, Th_5, Th_6, Th_7 et Th_8 correspondent à des seuils relatifs respectivement à :
  1. la hauteur minimale d'une composante connexe,
  2. les différences de position et de taille entre deux composantes en deçà desquelles un pixel placé entre ces deux composantes est dilaté lors de la dilatation conditionnelle,
  3. le rapport $\frac{largeur}{hauteur}$ minimal d'une boîte englobante,
  4. les taux de recouvrement mutuels minimums entre deux boîtes pour effectuer leur fusion.
- δ_X et δ_Y désignent la taille de l'agrandissement dans les directions x et y des boîtes englobantes lors de leur création pour compenser la différence entre les effets de l'érosion horizontale et ceux de la dilatation horizontale.

Il est intéressant de noter ici le nombre important de paramètres mis en jeu pour ce module de détection, paramètres auxquels sont associés des plages de variation quelques fois continues ($\alpha \in [0, 1]$). La taille de l'espace des paramètres motive ainsi le ciblage de l'optimisation proposée dans notre méthodologie.
Les parties suivantes détaillent alors les différents résultats obtenus en appliquant la méthodologie à ce module et ceci en prenant en compte les spécificités liées au texte présentées dans le chapitre précédent.

2.3 Analyse des performances du module de détection : extraction des comportements

Les expérimentations sont menées sur un journal télévisé (diffusé sur France 2 en Mai 2003) d'un durée de 37 minutes environ (soit pour un *framerate* de 25 images par secondes, 55500 images). Le choix d'un tel document repose sur deux facteurs. D'une part les journaux télévisés contiennent de nombreux textes ("de scène" ou artificiels), justifiant ainsi d'un point de vue quantitatif l'intérêt qui leur est porté. D'autre part, l'ensemble des journaux télévisés pour lesquels la charte graphique (en relation notamment avec la police , la couleur ou encore la taille des textes artificiels) demeure inchangée, forme une collection relativement homogène du point de vue des objets textes qu'elle contient, nous autorisant ainsi à envisager d'étendre l'utilisation des résultats de l'adaptation obtenus sur le journal choisi à cette collection entière.

Toutes les instances d'un même texte forment un unique objet spatio-temporel (un VideoText dans le formalisme adopté). Par ailleurs, nous rappelons ici que la vérité terrain est hiérarchique. Une fois construite, celle-ci contient :

- **543** VideoText "mot",
- **214** VideoText "ligne",
- **226** VideoText "bloc".

Le module considéré dans cette partie est celui de la détection. Toutes les instances d'un même texte doivent donc être considérées isolément, c'est à dire sans tenir compte de l'aspect temporel. Par ailleurs, étant donnée la nature des mesures d'évaluation choisies, qui prennent en compte les différents niveaux de représentation d'un texte (mots, lignes et bloc), il convient de s'appuyer sur l'ensemble de ces représentations. On parlera ainsi par la suite de la "famille" d'un texte pour désigner sa représentation comme ensemble d'un bloc, de lignes et de mots. En

conséquence, en prenant en compte les durées d'apparition, les différents VideoText de la vérité terrain regroupent 21514 instances de "*familles*" de texte différentes. Les évaluations porteront donc par la suite sur la détection de ces 21514 "*familles*".

Le module de détection produit 461035 zones de textes. Ce nombre paraît ici extrêmement défavorable au système comparé à celui de la vérité terrain. Il sera pour autant montré dans la suite qu'il convient de tempérer l'interprétation de ces résultats.

2.3.1 Extraction des oublis et des fausses alarmes

Il convient de ne pas pénaliser excessivement le système d'extraction de textes au regard du nombre de zones qu'il détecte et ceci pour deux raisons :

1. **La plupart des zones détectées correspondant à des fausses alarmes sont instables du point de vue temporel** et seront ainsi filtrées lors de l'étape de suivi (cf figures 2.2 et 2.3). Après la phase de suivi, le nombre d'instances de texte chute alors à 214761 ce qui induit une diminution drastique du nombre de fausses alarmes. On pourra aussi remarquer qu'une dernière étape de filtrage est envisageable à l'issue de la reconnaissance en s'appuyant sur des dictionnaires.

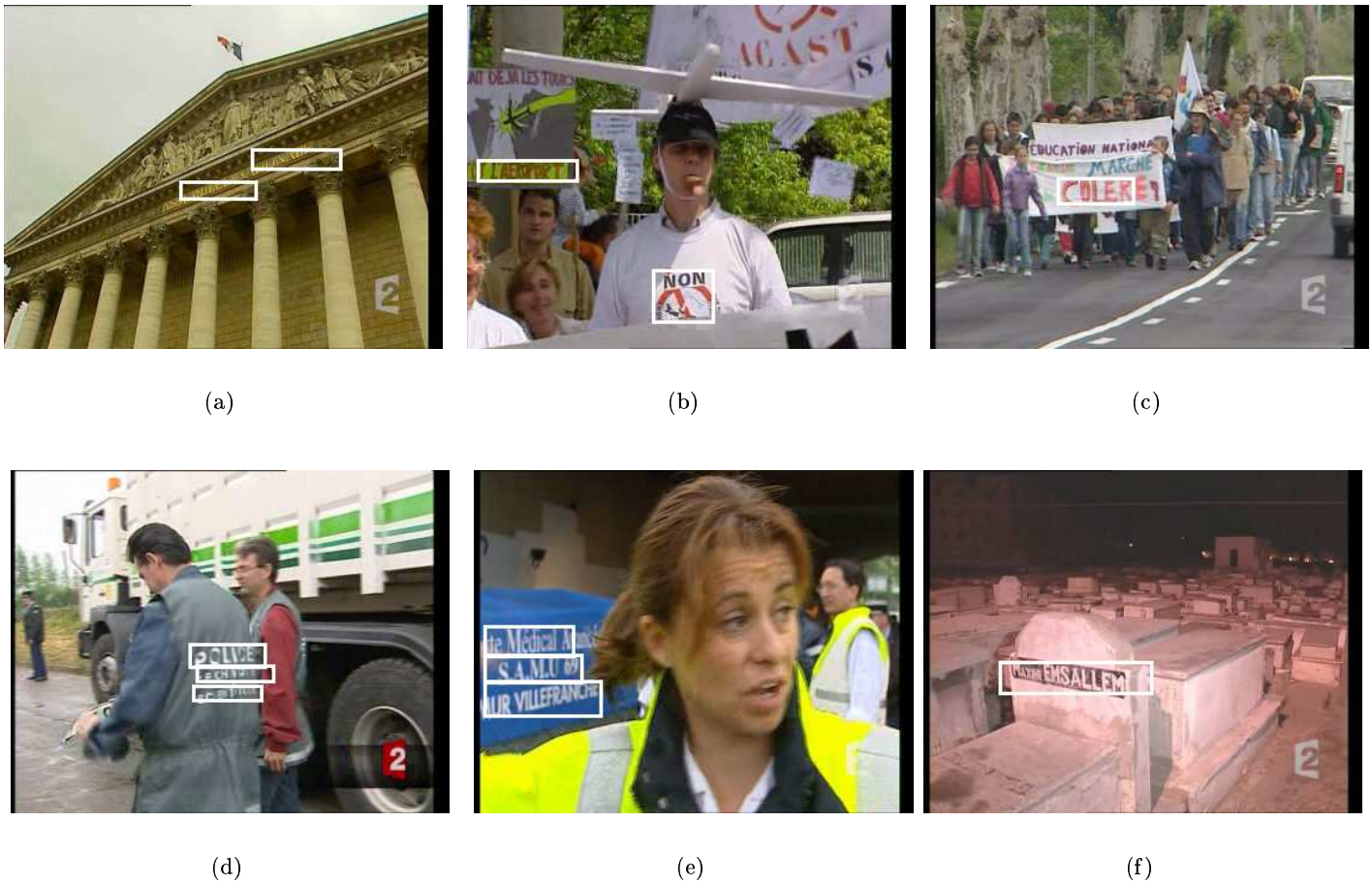
(a) Image 0 du flux (b) Image 1 du flux (c) Image 2 du flux

FIG. 2.2 – Une séquence illustrant l'instabilité des fausses alarmes

(a) Image 1 du flux (b) Image 2 du flux

FIG. 2.3 – Une séquence illustrant l'instabilité des fausses alarmes : 7 fausses alarmes sont supprimées lors de l'étape de suivi

2. **Le contexte visuel d'application du système lui est particulièrement défavorable**, puisque le journal choisi montre de nombreuses scènes de manifestations ou de regroupements qui sont généralement extrêmement texturées et provoquent ainsi de nombreuses fausses alarmes. En contrepartie, de tels évènements permettent dans le même temps d'évaluer le module de détection sur des textes particulièrement difficiles (typiquement les banderoles des manifestations). Il apparaît alors que le système évalué obtient des résultats impressionants sur certains textes de ce type (cf figure 2.4).



(a) (b) (c)

(d) (e) (f)

FIG. 2.4 – Des textes difficiles à détecter sur lesquels le module de détection est performant

L'extraction des oublis et des fausses alarmes repose sur les critères de recouvrement mutuels déjà cités. La vérité terrain est alors comparée avec les résultats du système selon différents seuils de recouvrement. L'évolution de la population de ces deux classes particulières est montrée dans la figure 2.5.

Une remarque intéressante concerne la courbe relative au taux de fausses alarmes : son aspect de courbe quasi-plane tend ainsi à montrer que les fausses alarmes générées par le module de détection sont des zones "*isolées*" et non des zones présentant une intersection avec certaines zones de la vérité terrain. En effet, si tel était le cas, il existerait un seuil concernant le recouvrement qui permettrait de ne plus considérer ces zones comme des fausses alarmes. Dans ce cas, la courbe présenterait plus de variations que présentemment. Ce constat témoigne une nouvelle fois de la difficulté à traiter le document choisi : l'aspect de la courbe suggère ainsi l'existence dans

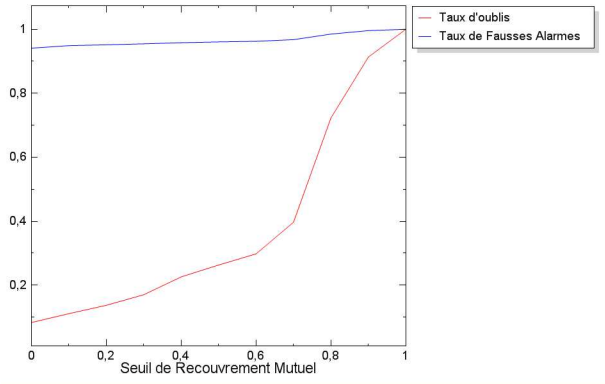


FIG. 2.5 – Evolution de la population des classes "*Oublis*" et "*Fausses Alarmes*" en fonction du seuil de recouvrement utilisé

les images que contient ce document de nombreuses zones dont l'aspect textural se rapproche de celui du texte (du point de vue du module de détection) et qui peuvent ainsi générer de nombreuses fausses alarmes.

Le choix du seuil utilisé pour isoler les oublis et les fausses alarmes relève alors de la volonté de l'utilisateur de pratiquer une adaptation plus ou moins importante. Le seuil de 0.2 (pour lequel les taux d'oublis et de fausses alarmes sont respectivement de 14% et 95%) est alors utilisé (on soulignera une nouvelle fois que ce taux très élevé de fausses alarmes n'est pas le reflet des résultats finaux obtenus par le système après application du module de suivi). Les images de la figure 2.6 montrent quelques zones de textes oubliées par le module de détection.

(a) (b) (c)

FIG. 2.6 – Quelques textes *oubliés* par le module de détection (dans l'image d, le texte est situé en bas à gauche : "11.04.2002")

2.3.2 Classification des comportements

Une fois extraits les comportements "*extrêmes*" que sont les oublis et les fausses alarmes, l'ensemble des éléments de la vérité terrain pour lesquels il est possible d'établir une association avec un (ou plusieurs éléments) issus des résultats, forme la base sur laquelle la classification va être appliquée. Cette base contient alors 18568 éléments (21514-2946 oublis)

Dans un premier temps, chacun de ces éléments se voit attribuer un vecteur de performances établi selon la méthodologie précisée dans le chapitre 1 de la partie 2 et le chapitre 1 de la partie 3. La phase de clustering repose alors sur la méthode précisée dans le chapitre 1 de la partie 2 dont sont donnés ci-dessous quelques rappels :

1. Le principe consiste à comparer les résultats de plusieurs méthodes de clustering,
2. Une méthode de linkage (clustering hiérarchique agglomératif (CHA)) est appliquée avec plusieurs paramétrages (type de normalisation des données, distances utilisées)
3. Le nombre de classes "*optimal*", C_{opti}, qui nous est inconnu, est évalué par une application itérative (sur le nombre de classes) d'un algorithme des k-moyennes,
4. Le résultat final est choisi entre :
   - le meilleur résultat de clustering de type CHA obtenu (parmi les résultats produits pour les différents paramétrages), coupé pour obtenir le nombre de classes C_{opti},
   - le meilleur résultat de clustering de type k-moyennes parmi un ensemble d'applications successives (la nature des centroïds initiaux varie à chaque itération) pour $k = C_{opti}$,
5. la détermination du meilleur résultat hiérarchique repose sur l'indice de Pearson (ou mesure de Cophenet) et la mesure d'évaluation d'un résultat de clustering est l'indice de Davies-Bouldin.

Les tableaux 2.8, 2.9 et 2.10 résument les résultats produits en appliquant cette méthode aux 18568 éléments à classer. Dans les tableaux 2.8 et 2.9, "Norma." désigne la normalisation des données appliquée, "Dist." désigne la distance utilisée entre les différents éléments, "Link." désigne la distance utilisée entre un élément (ou un cluster) et un cluster : S pour "*Single*", C pour "*Complete*" et A pour "*Average*". La distance "*Single*" correspond à la distance min, la distance "*Complete*" à la distance max et la distance "*Average*" à la distance moyenne (toujours entre l'élément (ou le cluster) et un cluster). Enfin "P." désigne l'indice de Pearson.

Norma.	$\sigma - \mu$																							
ACP	Oui												Non											
Dist.	L_1			L_2			L_3			L_∞			L_1			L_2			L_3			L_∞		
Link.	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A
P. $\times 10^{-3}$	83	80	92	78	90	92	75	77	91	71	82	86	73	79	89	78	90	92	81	84	93	86	79	**94**

TAB. 2.8 – Indices de Pearson des hiérarchies produites par les différents paramétrages du CHA

Norma.	Min-Max																							
ACP	Oui												Non											
Dist.	L_1			L_2			L_3			L_∞			L_1			L_2			L_3			L_∞		
Link.	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A	S	C	A
P. $\times 10^{-3}$	76	84	**90**	61	78	88	54	77	87	50	77	85	64	78	88	61	78	88	61	81	88	68	79	88

TAB. 2.9 – Indices de Pearson des hiérarchies produites par les différents paramétrages du CHA (suite)

L'analyse des tableaux 2.8 et 2.9 montre que le paramétrage optimal (on notera CHA_{opti} l'algorithme lui étant associé), qui correspond à celui associé à l'indice de Pearson le plus élevé, est le suivant :

1. Normalisation $\sigma - \mu$ sans ACP,
2. Distance L_∞,

3. Distance inter-cluster de type moyenne ("A").

Le nombre optimal de classes est estimé en appliquant l'algorithme des k-moyennes en faisant varier la valeur de k. De nouveau, pour plus de robustesse, différents "*pré-traitements*" des données sont pris en compte. La qualité des résultats produits pour chaque valeur de k est établie selon l'indice de Davies Bouldin produit et reportée dans le tableau 2.10 (on cherche l'indice le plus faible possible).

NORM.		Min-Max		$\sigma - \mu$	
ACP		oui	non	oui	non
CLASSE	2	1.01	1.01	1.30	1.38
	3	1.05	1.06	1.24	1.17
	4	0.98	1.16	***0.86***	0.88
	5	0.86	0.86	0.95	1.01
	6	***0.80***	***0.84***	0.87	0.86
	7	0.83	0.88	0.88	***0.82***
	8	0.99	0.87	0.94	0.94
	9	0.94	1	0.94	0.95
	10	0.95	0.85	0.97	0.81
	11	0.97	1.06	0.79	0.90
	12	0.99	0.95	0.94	0.90
	13	1.06	1	1.02	0.97
	14	0.95	0.98	0.89	0.90
	15	1.04	0.92	1.01	0.95
	16	0.99	0.97	1.08	0.94
	17	0.96	0.92	0.94	1.10
	18	1	0.96	0.95	0.94
	19	1.01	0.92	1.03	0.98
	20	0.95	0.94	1.01	0.95

	MinMax		$\sigma - \mu$	
	oui	non	oui	non
21	0.95	0.92	1.20	1.02
22	0.98	0.97	0.90	0.98
23	0.93	0.92	0.94	0.94
24	0.95	0.98	0.98	0.96
25	0.92	0.92	0.95	0.92
26	0.91	0.96	0.92	0.86
27	0.94	0.89	0.95	0.95
28	0.99	0.99	1.06	0.90
29	0.89	0.93	0.94	0.97
30	0.90	0.99	0.90	0.88

TAB. 2.10 – Résultats de l'évaluation du nombre de classes optimal par la méthode des k-moyennes

La valeur la plus faible de l'indice de Davies Bouldin est obtenue pour k=6 sur les données pré-traitées selon la méthode "Normalisation Min-Max avec ACP" (appelée par la suite *MMACP*).

Pour choisir le résultat final du clustering, l'arbre représentant la hiérarchie produite par CHA_{opti} est coupé pour obtenir 6 classes. D'autre part, l'algorithme des k-moyennes est appliqué avec k=6 sur les données pré-traitées selon la méthode MMACP, en faisant varier la position des centres initiaux des classes (1000 itérations) et le meilleur résultat obtenu (noté $Kmoy^{opti}$) est conservé.

Les scores obtenus sont les suivants :

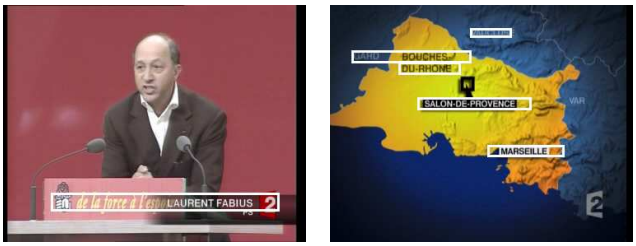
- $\mathbf{CHA_{opti}}$: 0.72,
- $Kmoy^{opti}$: 0.82.

Le score minimal est obtenu par la coupe de l'arbre produit par CHA_{opti}. La méthode de clustering utilisée pour traiter les vecteurs de performance produits consiste alors à appliquer le

clustering hiérarchique correspondant au paramétrage de CHA_{opti} et à couper l'arbre produit pour obtenir 6 classes.
Les images des figures 2.7 à 2.12 montrent alors des éléments issus de chacune des 6 classes de comportement produites selon cette méthode de clustering (dans ces images, les fausses alarmes n'apparaissent pas pour plus de clarté).

(a) (b) (c)

FIG. 2.7 – Images extraites de la classe 1



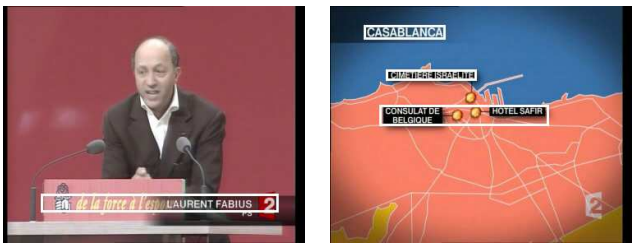
(a) Le texte concerné correspond au titre du pupitre "*de la force à l'espoir*"

(b) Le texte concerné est "GARD" en haut à gauche de l'image

FIG. 2.8 – Images extraites de la classe 2

(a) (b) (c)

FIG. 2.9 – Images extraites de la classe 3



(a) Le texte concerné est "Laurent Fabius (PS)" sur deux lignes

(b) Le texte concerné est "Hotel Safir" sur la droite de l'image

FIG. 2.10 – Images extraites de la classe 4

(a) (b) (c)

FIG. 2.11 – Images extraites de la classe 5

(a) (b) (c)

FIG. 2.12 – Images extraites de la classe 6

L'analyse de l'homogénéité des classes produites n'est pas aisée puisqu'elle nécessite d'avoir recours à une inspection visuelle des éléments les constituant. Il est tout de fois possible de donner quelques éléments d'interprétation pour chacune d'entre elles, sachant que dans les cas les plus complexes, la caractérisation repose sur des combinaisons des éléments composant les vecteurs de performances, dont il est impossible d'établir la nature visuellement.

1. **La classe 1** qui contient par ailleurs la très grande majorité des éléments (cf tableau 2.11), correspond aux cas pour lesquels la détection se déroule correctement. Les images extraites de cette classe (cf figure 2.7) montrent alors que le module de détection peut produire des résultats satisfaisants sur des textes de scène ("sapeurs pompiers" sur l'image a), sur des textes artificiels au niveau des lignes (image b) ou du bloc (image c).
2. **La classe 3** contient les cas pour lesquels la zone détectée par le système peut être simplement qualifiée de "trop grande" (dans ce cas, l'indice I^2_{recouv} est faible, pour un indice I^1_{recouv} élevé). On remarquera en analysant visuellement les images sur lesquels de tels résultats sont obtenus que ceux-ci interviennent lorsqu'il existe à proximité des textes, des zones présentant des propriétés texturales similaires à celles supposées des zones de textes (zones de textures verticales).
3. **Les classes 2 et 4** correspondent à des situations visuellement similaires. Pour autant, une analyse plus fine révèle les différences entre ces deux classes : la classe 2 concerne les textes détectés entièrement "*subissant*" une fusion ("*de la force à l'espoir*" par exemple) et la classe 4 contient quant à elle les textes subissant aussi une fusion (dans le cas de l'image a, il s'agit de la même fusion), aboutissant à de plus fortes dégradations de la qualité des résultats. Dans un des cas montrés, une unique ligne a été correctement détectée (la ligne "PS" n'est pas détectée dans l'image a). Dans l'autre cas, la fusion est pratiquée avec une zone de taille différente, détectée au niveau du bloc (le bloc "Consulat de Belgique" pour le texte "Hotel Safir" dans l'image b). Dans ce second cas, la différence de taille entre la zone détectée et la vérité terrain est ainsi beaucoup plus importante que dans le cas du texte "Gard" appartenant à la classe 2, qui fusionne avec une autre ligne ("Bouches").
4. **Les classes 5 et 6** contiennent des résultats pour lesquels la détection est effectuée au niveau des mots (à la différence de la majorité des résultats composant les autres classes). La seule différence apparente entre les deux classes est que le recouvrement global de la zone de la vérité terrain est inférieur dans le cas de la classe 6. Dans cette situation, l'analyse visuelle ne suffit pas à discriminer précisément les deux catégories de textes contenus dans ces classes.

Enfin, on remarquera que les textes de scène donnent lieu à des comportements plus variables. Plusieurs instances d'un même texte appartiennent ainsi à des classes différentes (par exemple le texte "*Parents avec Vous*" des images c et b des figures 2.12 et 2.11).

Une fois les classes de comportements extraites, l'enjeu est de les filtrer selon la valeur de la norme de leur centroïd en vue de sélectionner les seuls comportements insuffisants pour lesquels l'analyse des modules de niveau inférieur du module de détection devra être appliquée pour permettre d'établir la nature des paramètres qu'il convient de modifier. La tableau 2.11 montre alors les normes des centroïds des différentes classes obtenues ($\|\tilde{C}\|$) ainsi que leur cardinal.

Classe	*1*	*2*	*3*	*4*	*5*	*6*
Cardinal	18071	184	34	140	69	70
$\|\tilde{\mathbf{C}}\|$	2.60	1.82	1.58	1.24	1.61	0.74

TAB. 2.11 – Analyse des classes de comportements

Le tableau 2.11 confirme en partie l'analyse visuelle des classes puisque la classe 1 présente une norme élevée comparativement aux autres classes. On considérera par la suite les classes 2 à 6 comme des comportements insuffisants. Ces classes deviennent ainsi les *cibles* de l'adaptation dont les résultats sont donnés dans les parties suivantes.

2.4 Etablissement du diagnostic de responsabilité

2.4.1 Validation expérimentale du postulat mis en oeuvre

Comme expliqué dans le chapitre 2 de la partie 2, l'établissement du diagnostic de responsabilité repose sur le postulat suivant :

Postulat 3. *Chaque module de niveau inférieur entrant dans la composition de la séquence du module cible de l'adaptation, tend à améliorer les résultats obtenus par le module inférieur le précédant dans la séquence.*

Nous avons déjà donné une première validation de ce postulat basée sur son adoption dans d'autres travaux. Nous en proposons ici une validation expérimentale. Pour ce faire, il est nécessaire de considérer des cas dans lesquels le postulat est censé être vérifié, c'est à dire de considérer des images sur lesquelles le système de détection produit les résultats attendus.

Sept images, contenant des textes correctement détectés par le système sont donc choisies. Pour plus de clarté, le système est représenté ici comme la séquence de trois modules uniquement : le module textural, le module morphologique et le module géométrique (nous ne prenons donc pas en compte la décomposition de ces modules en séquences d'autres modules tel qu'illustré dans la figure 2.1).

Les performances des trois modules sont alors mesurées (à l'aide de la distance de Frobenius) sur ces sept images pour lesquelles la vérité terrain est construite. Ces performances sont illustrées dans la figure 2.13 tandis que la figure 2.14 montre quant à elle les différentes images considérées pour le calcul : la vérité terrain, tout comme les images produites par les trois modules considérés.

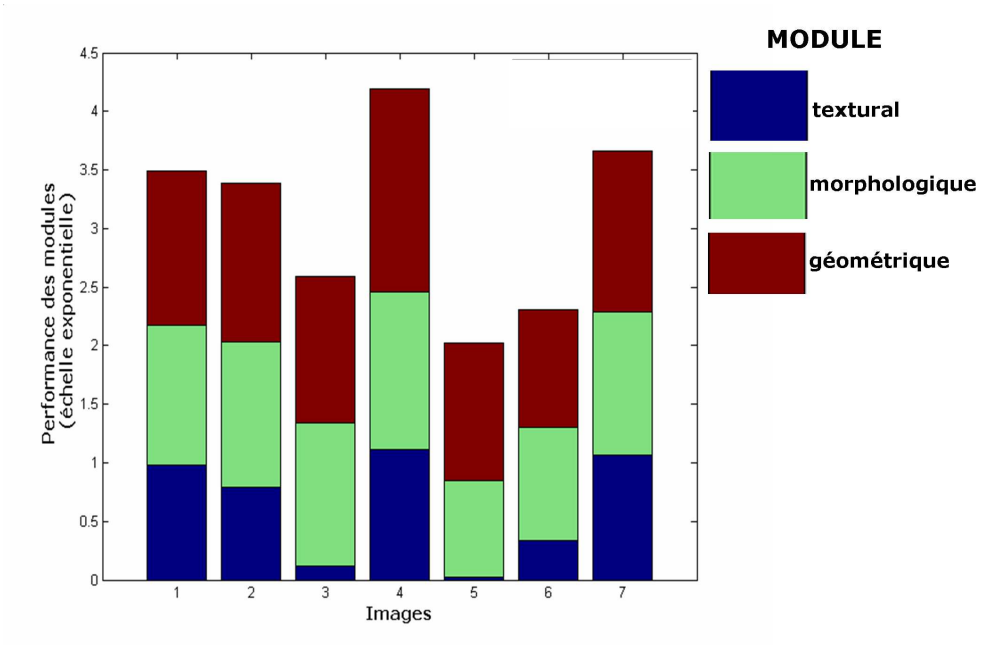
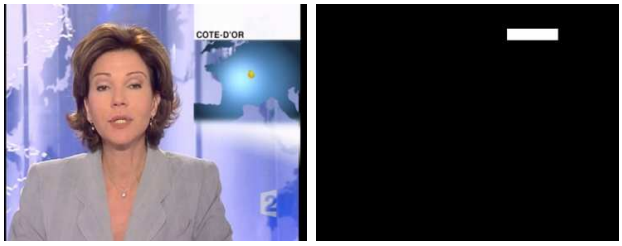


FIG. 2.13 – Performances des trois modules considérées mesurées sur les sept images choisies : la progression au cours de la séquence tend à valider le postulat adopté.



(a) Image originale

(b) Image de la vérité terrain : le texte est isolé

(c) Image produite par le module textural

(d) Image produite par le module morphologique

(e) Image produite par le module géométrique

FIG. 2.14 – Les images prises en compte lors du calcul des performances

L'analyse de la figure 2.13 révèle alors une progression des performances au fur et à mesure de l'application des modules de la séquence, et ceci pour les sept images considérées. Ce "rapprochement" avec la vérité terrain est d'ailleurs illustré visuellement dans les images de la figure 2.14. Cette analyse nous permet ainsi de valider expérimentalement le postulat adopté.

Dans la suite de cette partie, les résultats obtenus lors de la phase du diagnostic de responsabilité seront détaillés. Nous reviendrons ainsi sur la construction des bases dédiées, nécessaires au calcul des indices de responsabilité ; nous présenterons une analyse des courbes de performances produites sur ces bases et utiliserons ces mêmes courbes pour produire les indices de responsabilité et déterminer le module responsable.

2.4.2 Bases dédiées

Dans un premier temps, les bases d'images relatives aux différentes caractéristiques énumérées dans le chapitre précédent sont construites. Le nombre d'images contenues dans chacune de ces bases dépend de la caractéristique à laquelle celles-ci sont associées (cf tableau 2.12).

Caract.	*couleur1*	*couleur2*	*Contraste*	CT	CFV	CFH	OC_F	X_{pos}	Y_{pos}	L	H	O_T
#	128	128	192	90	18	34	90	102	133	24	83	36

TAB. 2.12 – Composition des bases dédiées

2.4.3 Analyse des courbes de résultats

Chacun des 12 modules composant la séquence du module de détection considéré (cf figure 2.1) est appliqué sur l'ensemble des images de chaque base pour produire les courbes de variation telles que celles montrées pour les modules du système de détection de visages dans le chapitre 2 de la partie 2. Etant donné que 12 caractéristiques sont considérées, on obtient donc un ensemble de 132 courbes. La figure 2.15 montre quelques unes de ces courbes en indiquant quel module et quelle caractéristique celles-ci concernent.

Relativement à l'interprétation de ces courbes, un premier point important concerne l'analyse des plages de variation de la qualité des résultats des différents modules. L'objectif final du module de détection est de produire des boîtes englobantes. La vérité terrain utilisée pour évaluer les résultats de chaque module est donc une image dans laquelle la zone englobante du texte est constituée de pixels marqués (leur niveau de gris est fixé ici à 255). La normalisation de la distance de Frobenius entre cette vérité terrain et les images produites par les différents modules[24] est effectuée relativement au nombre de pixels de textes contenus dans l'image de vérité terrain. Ainsi, dans le cas des fausses alarmes, cette distance peut être supérieure à 1, ce qui explique les quantités négatives de l'image (observées par exemple dans l'image c).
Le second point est un rappel : seules les variations des résultats des différents modules selon les valeurs d'une caractéristique sont analysées en vue de déterminer, pour chaque caractéristique, un module responsable. La fusion des diagnostics repose ensuite sur un vote à la majorité, l'ordre de mobilisation des différents modules responsables dans la séquence permettant le cas échéant de lever une ambiguïté dans ce vote. A aucun moment donc, une comparaison est effectuée entre les performances obtenues selon des caractéristiques différentes. L'analyse visuelle de ces courbes doit donc se limiter aux "allures" de celles-ci plutôt qu'aux valeurs des résultats obtenus. Par ailleurs, toute comparaison entre les résultats obtenus selon différentes caractéristiques eût été difficile étant donné qu'il n'existe pas une homogénéité suffisante entre les bases pour qu'une telle analyse soit envisageable.

Les courbes de la figure 2.15 rendent compte des *a priori* sur le comportement des modules, validant ainsi leur "allure" :

1. **Image a** : les performances du module de Sobel sont proportionnelles au contraste du texte avec le fond (plus les contours du texte sont marqués, plus la réponse à cet opérateur est forte). Par ailleurs, on remarque aussi que l'application du module d'accumulation permet de s'approcher de la vérité terrain dans la mesure où la réponse au module de Sobel est étendue selon la fenêtre d'accumulation utilisée.
2. **Image b** : le module de Gradient accumulé montre des performances moindres près des bords de l'image, reflet de la méthode utilisée pour le mettre en application. En effet, la fenêtre d'accumulation, lorsqu'elle est centrée sur les pixels situés près des bords de l'image, impose d'agrandir l'image d'autant de pixels que la position du pixel considéré l'impose (pour une fenêtre de taille 11 pixels et un pixel situé sur la première colonne de l'image (une position de type (0,j)), il convient de prendre en compte 5 pixels de valeur nulle situés "en dehors de l'image"). Les performances de ce module aux bords de l'image se voient donc diminuées .
3. **Image c** : l'augmentation de la complexité horizontale du fond diminue les performances

[24] On rappelle ici que l'image produite pour les modules géométriques est construite selon le même mode que celle de la vérité terrain, c'est à dire à partir des rectangles détectés.

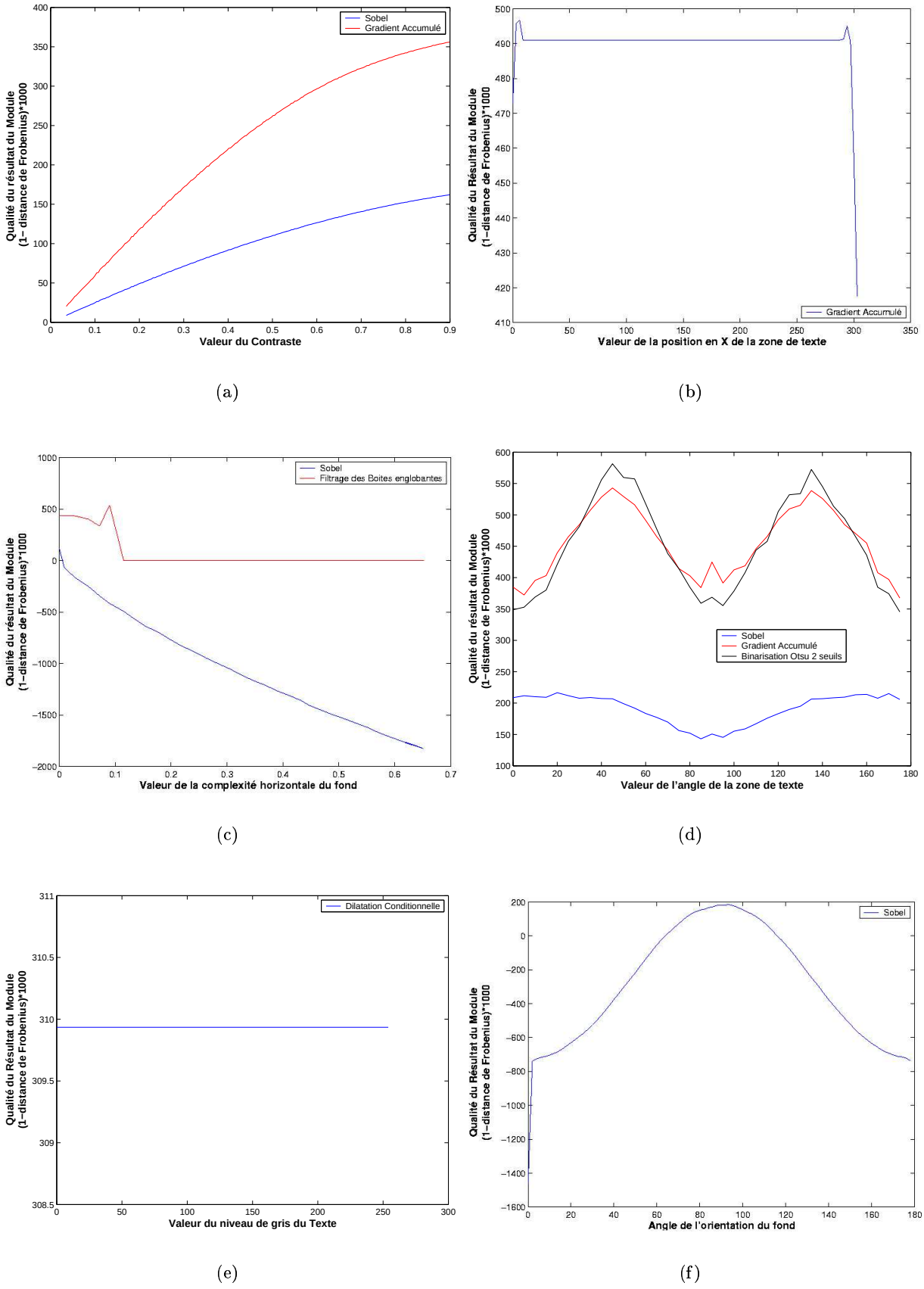
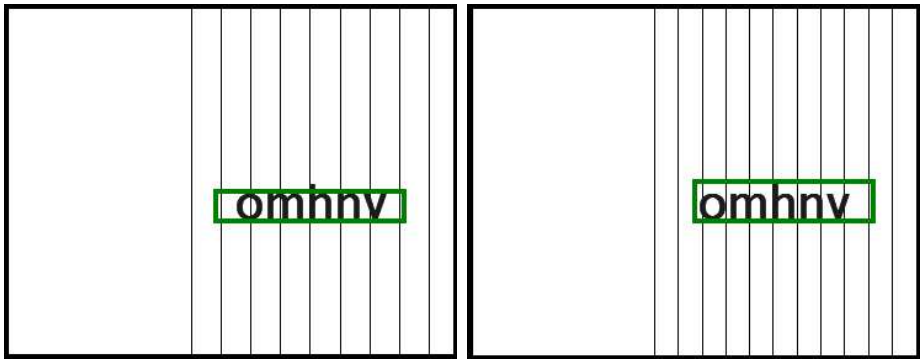


Fig. 2.15 – Courbes de variation de la qualité des résultats de certains modules selon différentes caractéristiques

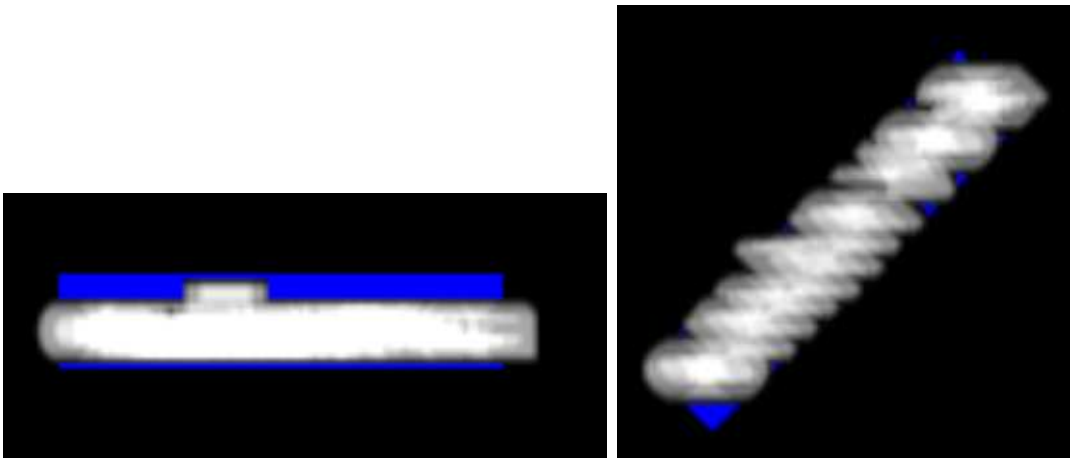
du module de Sobel : à mesure que la valeur de cette caractéristique augmente, sa réponse aux pixels du fond est de plus en plus élevée, aboutissant à de nombreuses fausses alarmes. Les boîtes englobantes créées ne remplissent alors plus les contraintes géométriques et sont fitrées par le module de filtrage géométrique. Le pic observé dans la courbe d'évolution de ce dernier module est dûe à une amélioration ponctuelle de la qualité des boîtes englobantes créée : l'augmentation de la complexité verticale permet l'inclusion de la *hampe* d'un caractère (cf figure 2.16).



(a) (b)

FIG. 2.16 – L'augmentation de la complexité horizontale permet une amélioration ponctuelle des résultats par l'intégration dans la zone détectée de la hampe d'un caractère (le "h")

4. **Image d** : relativement à l'angle de rotation du texte, le module de Sobel obtient logiquement ces meilleurs résultats pour les textes proches de l'horizontal (autour de 0 et 180 degrés). Concernant le module d'accumulation, des pics sont observés autour des angles 45 degrés et 135 degrés. En ces positions, la réponse au module de Sobel est toujours importante et l'orientation de la zone "expose" un nombre de pixels plus important à l'accumulation. Les résultats de l'accumulation sont ainsi plus "proches" de la zone délimitée par la vérité terrain que pour une position horizontale (cf figure 2.17). Quant au module de binarisation, il permet une amélioration des résultats du module d'accumulation autour des angles 45 et 135 mais aboutit à une dégradation autour des angles 0, 90 et 180. Le seuil haut de cette méthode de binarisation correspond au seuil proposé par Otsu assimilant les deux distributions des niveaux de gris recherchés (texte et fond) à des gaussiennes. Autour des angles 0, 90 et 180 degrés, ces deux distributions sont plus facilement séparables puisque la réponse des pixels de texte au module de Sobel est plus importante. La binarisation est ainsi plus précise et se rapproche plus du texte et non de sa zone englobante. La dégradation constatée est donc liée à la vérité terrain utilisée : si cette dernière était composée uniquement des pixels de texte et non de ceux inclus dans la boîte englobante, ces résultats ne seraient pas observés.
   Autour de 45 et 135 degrés au contraire, la binarisation aboutit à une amélioration des résultats qui peut s'expliquer par la plus grande homogénéité des résultats du gradient accumulés en termes de distribution de niveaux de gris pour ces orientations (cf figure 2.18). En effet, le module de binarisation met en oeuvre un second seuil (un seuil bas) et tous les pixels dont la valeur se trouve entre ce seuil et le seuil haut proposé par Otsu sont assignés au texte s'il existe un chemin constitué de pixels dont la valeur est supérieure au

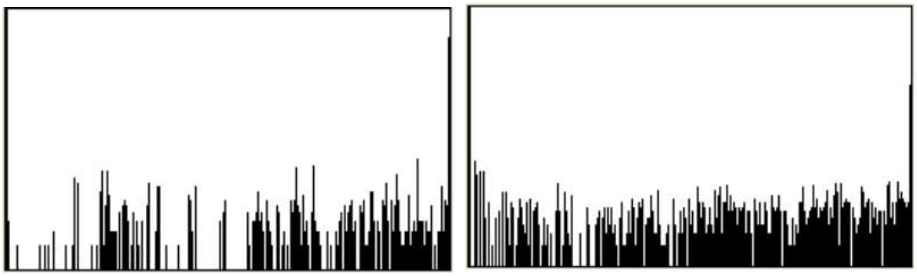


(a) Résultat du gradient accumulé pour un angle nul

(b) Résultat du gradient accumulé pour un angle de 45 degrés

FIG. 2.17 – Résultats du module de gradient accumulé pour des angles de 0 et 45 degrés. L'ensemble des pixels de la vérité terrain ayant une réponse nulle au module de gradient accumulé sont marqués en bleu : ils sont moins nombreux pour un angle de 45 degrés.

seuil bas, les reliant à un pixel de texte. L'homogénéité des niveaux de gris implique une proximité entre les deux seuils utilisés. Cette homogénéité implique dans le même temps une augmentation de la proportion des pixels dont la valeur se situe entre ces deux seuils pour lesquels il existe un chemin tel que précédemment décrit.
On notera aussi la symétrie de ces trois courbes autour de la valeur de 90 degrés.



(a) Angle = 0 degré

(b) Angle = 45 degré

FIG. 2.18 – Histogrammes (en échelle logarithmique) montrant les distributions des niveaux de gris pour les résultats du module de gradient accumulé pour 0 et 45 degrés.

5. **Image e** : le fonctionnement du module de dilation conditionnelle est invariant relativement à la couleur du texte. Etant donné que la couleur n'est pas utilisée comme caractéristique prédominante dans le modèle du texte adopté, ce résultat est logique.
6. **Image f** : les résultats du module de Sobel varient en fonction de l'orientation du fond. Un pic est obtenu pour 90 degrés, c'est à dire lorsque l'orientation du fond est proche de l'horizontale, auquel cas, le fond présente une réponse moindre au module de Sobel

horizontal. Au contraire, une dégradation des performances est constatée autour de 0 et 180 degrés (lorsque l'orientation du fond est proche de la verticale), valeur pour lesquelles la réponse des pixels du fond au module de Sobel horizontal est importante, provoquant ainsi des fausses alarmes. La même symétrie que celle constatée pour l'image d, autour de la valeur de 90 degrés, est par ailleurs observée.

2.4.4 Détermination des modules responsables

Pour chacune des 5 classes de comportements considérés pour l'adaptation, l'analyse des courbes permet de restreindre l'optimisation aux seuls modules diagnostiqués responsables de l'erreur constatée. Dans un premier temps, les plages de variation de chaque caractéristique, dont la connaissance est nécessaire au calcul des indices de responsabilité, sont calculées (cf tableau2.13).

	Classe 2		Classe 3		Classe 4		Classe 5		Classe 6	
	Min	*Max*	*Min*	*Max*	*Min*	*Max*	*Min*	*Max*	*Min*	*Max*
couleur1	59	157	36	167	25	66	42	150	40	101
couleur2	101	217	143	238	144	192	160	208	158	191
contraste	0.003	0.02	0.0018	0.12	0.005	0.15	0.0010	0.009	0.00014	0.0073
CT	0.0004	0.012	0.0004	0.008	0.0006	0.002	0.0001	0.005	0.0002	0.0045
CFV	0.03	0.11	0.025	0.20	0.09	0.22	0.036	0.087	0.022	0.065
CFH	0.05	0.15	0.05	0.17	0.10	0.18	0.039	0.095	0.036	0.058
OC_F	26	86	14	74	26	86	26	74	44	74
positionX	37	135	135	313	102	250	132	219	170	211
positionY	60	238	25	248	127	246	58	249	48	245
largeur	41	131	44	259	69	142	111	290	117	314
hauteur	19	38	16	59	17	29	30	120	32	158
O_T	0	56.25	0	90	0	56.25	0	78.75	0	67.5

TAB. 2.13 – Plages de variations des caractéristiques sur les 5 classes de comportements considérées pour l'adaptation

L'analyse de ce tableau révèle que pour certaines caractéristiques (le contraste et les "*complexités*"), les plages sont extrêmement étroites au regard des plages manipulées sur les bases dédiées (entre 0 et 1 pour ces 4 caractéristiques). Avant même d'effectuer le diagnostic de responsabilité, il apparaît donc que la précision des bases peut être insuffisante pour rendre compte de variations de comportements sur des plages aussi réduites. Pour l'instant, les bases sont constituées d'images artificielles et il est difficile d'obtenir selon ce mode de construction une meilleure précision, c'est à dire un contrôle plus fin des variations des caractéristiques sur les images créées. La solution la plus envisageable consiste alors à utiliser des images réelles issues du flux audiovisuel considéré. Malheureusement, pour la construction d'une base donnée, relative à une certaine caractéristique, cette solution permet de gagner en précision, mais entraîne dans le même temps une perte au niveau du contrôle des variations parasites des caractéristiques non prises en compte. La constitution des bases selon ce mode impliquerait donc de prendre en compte ces variations parasites lors du calcul des indices de responsabilité. Ceci constitue ainsi une évolution intéressante de la méthodologie telle qu'elle existe actuellement. Pour remédier au manque de précision des bases, une interpolation des courbes selon des splines cubiques est effectuée.

Etant donné les plages obtenues, les indices de responsabilité sont calculés et un module responsable est désigné pour chaque classe et chaque caractéristique (celui obtenant l'indice minimal parmi l'ensemble des modules). Le tableau 2.14 montre alors les diagnostics établis dans chaque cas, le diagnostic établi par le vote à la majorité ainsi que le diagnostic final, différent du diagnostic par vote en cas d'ambiguïté. Les modules sont nommés ici selon leur ordre de mobilisation dans la séquence du module de détection : (M_0, Sobel), (M_1, Gradient Accumulé), (M_2, Binarisation Otsu 2 seuils), (M_3, Fermeture), (M_4, Suppression des ponts), (M_5, Dilatation Conditionnelle), (M_6, Erosion Conditionnelle), (M_7, Erosion Horizontale), (M_8, Dilatation Horizontale), (M_9, Création des boîtes englobantes), (M_10, Filtrage des boîtes englobantes), (M_{11}, Fusion des boîtes englobantes).

		Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
diagnostic	*couleur1*	M_2	M_2	M_0	M_0	M_0
	couleur2	M_2	M_2	M_2	M_2	M_2
	contraste	M_0	M_8	M_0	M_9	M_9
	CT	M_{10}	M_{10}	M_7	M_{10}	M_{10}
	CF_{Horiz}	M_3	M_2	M_2	M_3	M_3
	CF_{Verti}	M_3	M_3	M_3	M_{10}	M_5
	OC_F	M_0	M_0	M_0	M_1	M_{10}
	positionX	M_3	M_3	M_5	M_5	M_5
	positionY	M_1	M_1	M_1	M_1	M_1
	largeur	M_5	M_8	M_5	M_5	M_5
	hauteur	M_2	M_2	M_2	M_7	M_7
	O_T	M_6	M_6	M_6	M_6	M_6
	VOTE	M_2 ou M_3	M_2	M_0 ou M_2	M_1, M_5 ou M_{10}	M_5
	FINAL	$\mathbf{M_2}$	$\mathbf{M_2}$	$\mathbf{M_0}$	$\mathbf{M_1}$	$\mathbf{M_5}$

TAB. 2.14 – diagnostics de responsabilité obtenus sur les différentes classes de comportement pour les différentes caractéristiques

L'analyse du tableau 2.14 donne alors lieu aux remarques suivantes :

- le diagnostic associé à certaines caractéristiques est le même quel que soit la classe considérée : c'est le cas pour la position en Y de la zone de texte ainsi que pour l'angle de rotation du texte. Deux facteurs peuvent expliquer ce résultat : d'une part, la proximité des plages de variation des caractéristiques pour les différentes classes, et d'autre part le fait que la taille de ces plages soit relativement large au regard de la plage de la base dédiée (ce qui réduit la variabilité des indices pouvant leur être associés).
- pour chaque classe, certains modules ne sont jamais déclarés responsables (cf tableau 2.15), ce qui réduit ainsi dans l'absolu le nombre de paramètres à prendre en compte pour l'optimisation. La méthodologie montre donc ici son caractère modulaire : il est ainsi possible de limiter la restriction de l'ensemble de paramétrage aux paramètres des modules déclarés responsables pour chaque caractéristique après le calcul des indices de responsabilité, sans tenir compte du vote final.
- l'assignation de la responsabilité au module de Sobel (M_0) pour la classe 4 soulève la question d'une remise en cause de la structure du module de détection. En effet, le module de Sobel ne possède pas de paramètres. En conséquence, le fait qu'il soit déclaré responsable implique soit de changer d'opérateur pour le calcul du gradient ou alors de modifier plus en profondeur le système en intégrant un détecteur différent (par exemple

	Classe 2	**Classe 3**	**Classe 4**	**Classe 5**	**Classe 6**
	M_4	M_4	M_4	M_3	M_4
	M_7	M_7	M_8	M_4	M_8
	M_8	M_9	M_9	M_8	M_{11}
	M_9	M_{11}	$M_1 1$	M_{11}	
	M_{11}				
Paramètres	8	7	7	7	6

TAB. 2.15 – Modules déclarés responsables pour aucune analyse liée à une caractéristique

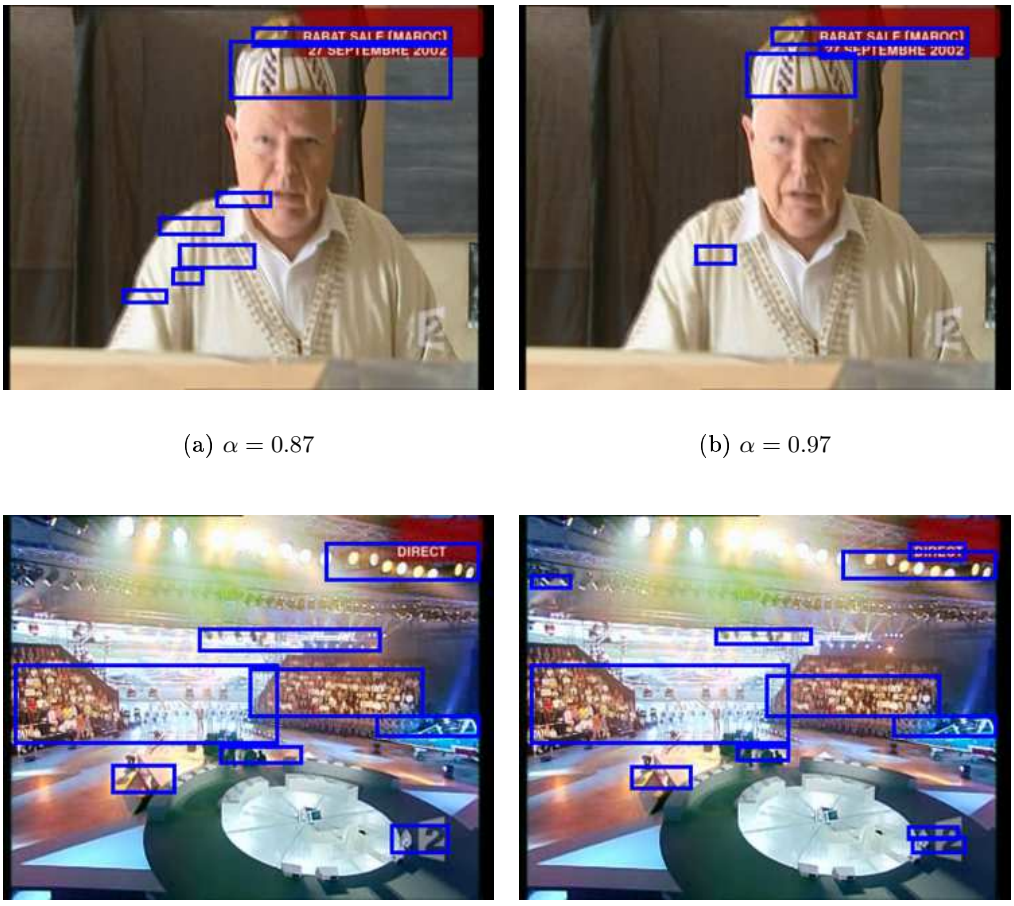
un détecteur de coins, un détecteur de Sobel généralisé $k \times k$ avec k>3, ou encore un opérateur de Canny-Deriche dont le seul paramètre α est lié à la notion de résolution). On constate donc ici que la méthodologie permet de mettre en avant des limitations du systèmes autres que celles uniquement liées au paramétrage.

Pour valider les diagnostics obtenus, l'idée est alors de pratiquer l'optimisation des paramètres des modules désignés responsables pour chaque classe et de vérifier que la modification de ces paramètres permet d'obtenir de meilleurs résultats. Nous limitons ici l'analyse aux classes 3 et 5. Dans le cas de la classe 3, le module déclaré responsable est le module de binarisation (M_2) auquel est attaché le paramètre α, réglant le second seuil utilisé pour la binarisation. Pour ce qui est de la classe 5, c'est le module d'accumulation (M_1) qui est désigné responsable. Ce module comporte lui aussi un unique paramètre, S, qui désigne la taille de la fenêtre d'accumulation. Les valeurs initiales de ces deux paramètres sont respectivement 0.87 et 13.

L'idée est alors de faire varier ces deux paramètres et de vérifier l'incidence positive de cette variation sur les résultats produits. Les figures 2.19 et 2.20 montrent alors les résultats obtenus sur des images issues respectivement de la classe 3 et de la classe 5 pour différentes valeurs des paramètres concernés. Nous remarquerons ici la présence de fausses alarmes dans les images produites, contrairement aux images montrées dans les figures 2.7 à 2.12. Néanmoins, ces fausses alarmes ne doivent pas être prises en compte pour analyser les résultats puisque le filtrage des fausses alarmes n'a pas encore eu lieu (les résultats obtenus par la méthode de filtrage développée seront détaillés dans la partie suivante). Seule la précision de la détection des zones de texte doit donc être considérée.

L'analyse des résultats produits pour les deux images issues de la classe 3 montre que la modification du paramètre lié à la binarisation permet une amélioration des performances : en choisissant pour α une valeur proche de 0.96/0.97 on observe une réduction de l'erreur obtenue pour le paramétrage initial, à savoir une zone de détection trop grande.
Les mêmes conclusions peuvent être menées au regard des images montrées dans la figure 2.20. Ces images montrent en effet qu'une amélioration commune aux deux images est obtenue en fixant la taille de la fenêtre d'accumulation S à 17. Le meilleur résultat obtenu pour la première image (première ligne) est obtenu pour S = 19. Néanmoins, l'objectif de l'optimisation est de trouver un unique paramétrage pour l'ensembles des zones de texte constituant le comportement considéré. La valeur S=17 semble donc ici convenir. Elle permet par exemple dans la seconde image de détecter le mot "VOUS" situé à droite sur la banderole, tout en améliorant le niveau de détection pour les autres mots ; les résultats obtenus sur la première image ne sont quant à eux pas dégradés.

Cette analyse nous a permis de montrer que la variation des paramètres des modules désignés responsables lors du diagnostic de responsabilité permettait d'améliorer les résultats du système de détection utilisé. Des expérimentations supplémentaires en vue de valider plus avant l'analyse du diagnostic seront proposées en conclusion de ce manuscrit. Reste uniquement, pour conclure cette partie consacrée aux expérimentations, à détailler les résultats obtenus par la méthode de filtrage des fausses alarmes que nous avons définie.

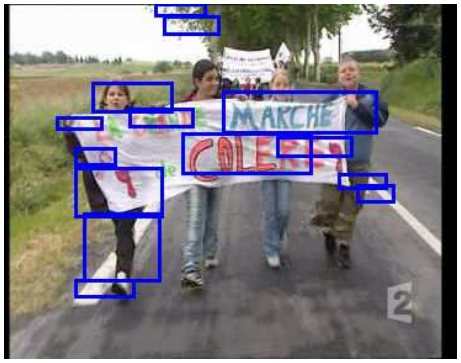


(a) $\alpha = 0.87$ (b) $\alpha = 0.97$

(c) $\alpha = 0.87$ (d) $\alpha = 0.96$

FIG. 2.19 – Optimisation sur quelques images isssues de la classe 3 : les images de la colonne de gauche montrent les résultats obtenus avec le paramétrage initial de α et les images de la colonne de droite montrent les résultats produits pour un paramétrage différent

(a) S=13



(b) S=17

(c) S=19

(d) S=13

(e) S=17

(f) S=19

FIG. 2.20 – Optimisation sur quelques images isssues de la classe 5 : les images de gauche correspondent aux résultats obtenus avec le paramétrage initial de S, les deux autres colonnes montrent des résultats produits avec des valeurs différentes pour ce paramètre.

2.5 Le traitement des fausses alarmes

Le traitement des fausses alarmes repose sur la définition de filtres permettant de supprimer *a posteriori* (c'est à dire après application du module de détection) ces erreurs. L'objectif de ces filtres est de supprimer les fausses alarmes tout en conservant les résultats corrects. Le principe est ainsi de construire une signature de ces deux ensembles : un élément détecté est alors considéré comme une fausse alarme (et donc supprimé) si sa signature est plus proche de celle de l'ensemble des fausses alarmes.

Nous avons détaillé dans le chapitre précédent le choix des caractéristiques utilisées pour produire les signatures. De plus, en nous appuyant sur un ensemble d'apprentissage (2000 éléments de la classe "Fausse alarme" et 2000 éléments de la classe "Résultats corrects"), la méthode de Fisher a été appliquée pour produire un vecteur de classement des caractéristiques en fonction de leur capacité à discriminer les deux classes considérées.
L'objectif est de produire les meilleures signatures possibles des deux classes, c'est à dire celles permettant de maximiser le taux de suppression des fausses alarmes tout en minimisant celui des résultats corrects. La solution immédiate consiste à calculer, pour différents ensembles de caractéristiques (choisis selon le vecteur de Fisher : première caractéristique seule, deux premières caractéristiques, ...), les signatures des deux classes (qui sont les centroïds respectifs de ces classes) et à choisir l'ensemble produisant les meilleures signatures au regard du critère précédent.
Nous proposons ici d'affiner cette solution en considérant pour la classe fausse alarme et pour la classe des résultats corrects, quatre sous-classes [25]. L'idée est ainsi de s'affranchir de l'hétérogénéité des données d'apprentissage en construisant pour les fausses alarmes et les résultats corrects quatre signatures sur des sous-ensembles plus homogènes. Pour produire ces classes, un algorithme de clustering des k-moyennes est utilisé. Plusieurs (24 exactement) ensembles de caractéristiques, toujours construits en fonction du vecteur de Fisher, sont utilisés et nous obtenons ainsi autant de résultats de clustering sur les deux ensembles (fausses alarmes et résultats corrects).
Pour chaque résultat de clustering, huit signatures sont donc calculées (4 pour les fausses alarmes et 4 pour les résultats corrects). Le filtrage est alors adapté de la façon suivante : un élément détecté est considéré comme une fausse alarme si la distance minimale entre sa signature et les huit signatures considérées est obtenue avec une signature d'une sous classe de type fausse alarme.

Pour évaluer les filtres produits, deux ensembles de test (2000 éléments de type fausse alarme et 2000 éléments de type résultat correct) sont construits. Les résultats sont alors évalués en termes de taux de suppresion des fausses alarmes et des résultats corrects. Ces expérimentations montrent alors que le meilleur résultat (cf tableau 2.16) est obtenu en utilisant l'ensemble des 24 caractéristiques. Ceci tend donc à prouver qu'il est difficile de distinguer les fausses alarmes des résultats corrects et que l'ensemble des caractéristiques considéré jusqu'à présent pourrait gagner à être étendu.

Les images de la figure 2.21 illustrent trois résultats corrects classés parmi les fausses alarmes. Ces textes sont complexes au regard de leur orientation, des cas d'occlusion subis ou encore de leur résolution. Ils correspondent ainsi à des cas assez rares relativement à l'ensemble des résultats corrects qui concerne en majorité des textes artificiels. La faible représentation de tels textes dans l'ensemble d'apprentissage des résultats corrects peut donc expliquer cette erreur

[25] le chiffre de quatre est choisi ici expérimentalement

Classe	*Fausse Alarme*	*Résultat Correct*
Taux de Suppression	0.67%	0.22%

TAB. 2.16 – Taux de suppresion des différents éléments des ensembles de test (pour 24 caractéristiques)

de classification.

(a) (b) (c)

FIG. 2.21 – Trois résultats corrects de l'algorithme classés parmi les fausses alarmes

Malgré quelques erreurs, ces premiers résultats sont encourageants. Il sera à l'avenir envisagé de les améliorer en utilisant des caractéristiques supplémentaires (comme par exemple des moments statistiques liés à la caractérisation des textures) et en prenant en compte des ensembles d'apprentissage plus vastes et "mieux" construits (en termes de représentativité des différentes catégories de résultats corrects par exemple).

2.6 Conclusion

Les résultats obtenus sur l'objet texte sont prometteurs et justifient l'utilisation de la méthodologie d'adaptation proposée. Bien que le module de détection du texte vidéo utilisé soit relativement complexe au niveau de la chaîne des traitements qui le constitue ; bien que les comportements observés, notamment sur certains textes de scène, soient particulièrement difficiles à interpréter, les différentes analyses développées permettent d'appréhender le fonctionnement de l'algorithme. Plus précisément, nous avons constaté que la méthodologie permettait de dresser un diagnostic de responsabilité valide, relatif aux différentes catégories d'erreur extraites.

Ce résultat encourageant constitue alors une motivation pour continuer dans cette voie et envisager des améliorations futures, améliorations qui seront détaillées dans la conclusion finale qui suit.

Quatrième partie

Conclusion et perspectives

1

Réussites et limites de la méthodologie : du travail fourni au travail à fournir

Sommaire

1.1 Résumé de la méthode développée

Le mécanisme d'une nouvelle méthodologie d'adaptation des systèmes d'extraction d'objets dans les flux audiovisuels (systèmes *DSRO*), dont l'apport essentiel se situe dans la préparation à la phase "*classique*" d'optimisation des paramètres a été exposée dans ce manuscrit.

La nécessité d'un tel méta-système a été dans un premier temps motivée par des considérations globales, sur les systèmes de vision en général et les systèmes *DSRO* en particulier, considérations portées principalement sur leur mode de conception. L'indéterminisme de tels systèmes, lié au fait qu'ils sont généralement associés à la résolution d'un problème mal défini, a ainsi été souligné. Par ailleurs, la difficulté de leur conception a aussi été mise en relation avec ce qui peut être appelé la *combinatoire de composition* : chaque concepteur doit choisir parmi un ensemble de modèles mathématiques, d'opérateurs adaptés aux traitements désirés, ... et la multitude de combinaisons qui s'offrent à lui rendent le choix de la composition finale difficile. Concernant les systèmes DSRO, la conception s'articule autour de la construction d'un modèle auquel est associé un filtre qui constitue la part algorithmique du système. La diversité des contenus ne permet pas de définir de modèles suffisamment génériques. Par ailleurs, quelque soit le modèle adopté, la construction d'un filtre associé robuste s'avère tout aussi délicate. Les résultats du système sont alors fonction de la distance entre les objets du corpus d'application et ceux utilisés lors de sa conception. L'adaptation des systèmes devient alors nécessaire pour maintenir un égal niveau de performance quelque soit le corpus concerné.

La méthodologie proposée s'appuie alors sur un unique niveau d'abstraction : celui des paramètres. Le découpage de tout système DSRO en une séquence de modules : Détection,

Suivi, Amélioration et Reconnaissance, permet d'adopter une vue séquentielle de l'adaptation. Au niveau de chaque module, l'idée motrice est alors de restreindre la phase d'optimisation aux seuls paramètres du module de niveau inférieur responsable de l'erreur constatée. A la différence d'une méthode "*aveugle*" d'optimisation pour laquelle tous les paramètres varient jusqu'à obtention d'un meilleur résultat, le ciblage proposé ici nécessite de comprendre le fonctionnement du module.

La méthodologie se base sur la fusion de deux analyses. La première porte sur les résultats produits par le système sur un corpus appelé corpus d'adaptation, corpus représentatif de l'ensemble des documents sur lesquels on souhaite appliquer le système. Cette analyse a pour objet l'extraction des différents comportements obtenus par ce dernier, c'est à dire des différentes catégories d'objets sur lesquelles le système produit des performances différentes. L'objectif final est alors de déterminer quels sont les comportements pour lesquels une adaptation est nécessaire, comportements appelés "comportements insuffisants".
La seconde analyse s'applique quant à elle aux modules de niveau inférieur composant la séquence du module considéré et a pour objet d'établir leur dépendance relativement à un ensemble de caractéristiques pré-déterminé. C'est finalement la représentation des comportements extraits par la première analyse selon ces mêmes caractéristiques qui permettra d'établir, pour chaque comportement "*insuffisant*", la nature du *module responsable*.

Concernant le module de la détection, les erreurs les plus souvent considérées sont les oublis ainsi que les fausses alarmes. Si le traitement des oublis repose sur la même méthodologie, un mécanisme différent a été développé pour les fausses alarmes, aboutissant à la création de filtres permettant de supprimer ces erreurs *a posteriori*, c'est à dire après l'application du module de détection.

1.2 Apports de la méthodologie

La contribution majeure de ce travail réside dans l'approche de l'adaptation pour laquelle nous avons opté, basée sur une tentative de compréhension des systèmes manipulés. En essayant de répondre automatiquement à la question : "Pourquoi ce système obtient-il des résultats insuffisants sur cette image ?", l'objectif, sans doute trop ambitieux, était de substituer la machine au concepteur dans le cycle "essais-corrections" de mise au point d'un système. En prenant en compte un minimum de connaissances *a priori*, ce travail a permis de développer les bases d'une nouvelle méthodologie de contrôle des systèmes DSRO, dite "*autonome*".

Il est important de faire ici une distinction entre les rouages principaux de la méthodologie, c'est à dire l'analyse des comportements et le diagnostic de responsabilité, et les technologies mises en oeuvre pour instancier cette méthodologie. Ainsi, si l'organisation de l'adaptation autour de deux analyses distinctes peut être considérée comme la principale originalité de ce travail, les technologies utilisées pour son instanciation peuvent être facilement remplacées. Certaines propositions seront ainsi effectuées sur quelques points précis, dans la partie portant sur les perspectives.

Enfin, outre les mesures d'évaluation proposées, un dernier point intéressant réside dans la méthode, malheureusement abandonnée, d'étude de la sensibilité des modules selon le principe de Helmholtz. Si elle nécessite une modélisation des modules complexes, elle n'en demeure pas

moins plus originale que la méthode effective finalement choisie.

1.3 Résultats obtenus et perspectives

La validité des choix effectués pour concevoir la méthode a été éprouvée au cours du manuscrit par l'analyse de l'objet "*visage*" en mettant en oeuvre un module de détection existant ([SP96]). La partie propre aux expérimentations repose quant à elle sur l'objet *texte* associé au module de détection du système DSRO dédié à cet objet exposé dans [Wol03].

Les résultats obtenus motivent alors d'autres travaux à plus ou moins long terme permettant de continuer à valider la méthodologie ou encore à l'améliorer.

1.3.1 A court terme

Concernant l'objet *texte*, les expérimentations menées ont donné les premiers indices de l'efficacité de la méthodologie : pour le traitement des comportements insuffisants extraits et pour le traitement des fausses alarmes. Il serait intéressant de traiter par la suite le cas des oublis, mais aussi de proposer une méthodologie d'évaluation quantitative du diagnostic de responsabilité.

Concernant les oublis, leur traitement repose sur les mêmes modalités que celui des comportements insuffisants, à ceci près que la variabilité de la classe des oublis en termes de caractéristiques visuelles peut aboutir lors du calcul des indices de responsabilité, à des plages de variations trop importantes pour permettre l'obtention d'un indice fiable. Une solution envisageable consiste alors à pratiquer un clustering sur les zones correspondant aux oublis, relativement aux caractéristiques visuelles choisies. Les classes obtenues bénéficieraient alors d'une plus grande homogénéité selon ces caractéristiques, facilitant le calcul des indices de responsabilité.

L'évaluation des diagnostics établis est plus complexe, puisqu'elle repose sur l'optimisation des résultats sur les différentes classes de comportements, en prenant en compte les seuls paramètres du module désigné responsable. Cette optimisation se doit d'être globale dans le sens où un unique paramétrage doit permettre l'amélioration de la qualité moyenne des performances sur tous les objets composant le comportement. Par ailleurs, si l'on admet qu'un tel paramétrage existe, la validation du diagnostic proposé impose de comparer l'amélioration des performances obtenue avec celles produites par l'optimisation des autres modules ou par celle de l'ensemble des modules. La restriction à laquelle aboutit le diagnostic est ainsi exploitable si la seule amélioration comparable est obtenue par l'optimisation d'un nombre important d'autres modules, comprenant d'ailleurs le module désigné responsable.
A l'issue de cette phase d'optimisation, il existe alors autant de paramétrages optimaux qu'il existe de comportements (un paramétrage optimal par comportement insuffisant auquel s'ajoute le paramétrage initial, associé au comportement dit "correct"). Il convient alors de chercher à utiliser l'ensemble de ces paramétrages et ceci sous la forme d'une intégration de l'ensemble des résultats produits en appliquant le système selon chacun d'entre eux. La mise en place de cette intégration constitue donc une perspective aux expérimentations déjà menées.
Par ailleurs, les expérimentations détaillées dans le manuscrit portent sur le seul module de détection. L'application de l'adaptation, selon la vue séquentielle adoptée, aux autres modules (Suivi, Amélioration et Reconnaissance) devra aussi par la suite être entreprise.

Enfin, il a été expliqué que la composition des bases de documents à l'Institut National de l'Audiovisuel se prêtait particulièrement à l'adaptation. La question est alors de savoir dans quelle mesure les paramétrages optimaux produits pour un document particulier d'un telle base permettent l'amélioration des performances du système sur l'ensemble des autres documents de la base considérée.

1.3.2 A plus long terme

Une première amélioration possible des technologies utilisées dans la méthodologie, concerne, comme nous l'avons déjà évoqué plus haut, la méthode, abandonnée, d'étude de la sensibilité des modules selon le principe de Helmholtz.

Une seconde perspective concerne la conception des bases utilisées pour l'établissement du diagnostic de responsabilité. Nous avons déjà expliqué que la conception, manuelle, de ces bases est particulièrement problématique car elle restreint la nature des caractéristiques qu'il est possible de prendre en compte, tout comme la précision des plages de variation des caractéristiques retenues. La possibilité d'utiliser des images réelles issues du flux vidéo à déjà été évoquée et constitue la piste la plus intéressante. Le problème relatif à ce mode de conception réside dans l'apparition de variations parasites de caractéristiques auxquelles la base considérée n'est pas dédiée.
Un premier objectif consiste alors à minimiser au maximum ces variations en choisissant les images relatives à chacune des bases de façon adéquate. Par la suite, il convient d'adapter la méthodologie pour pouvoir prendre en compte les variations résiduelles. L'adaptation de la méthodologie pourrait alors intervenir à deux stades différents : soit lors du calcul de l'indice de responsabilité ; soit lors la fusion des diagnostics établis pour chaque caractéristique, en attribuant par exemple un degré de fiabilité aux diagnostics selon la caractéristique (et donc la base) à laquelle ils sont attachés.

Une autre perspective intéressante concerne la capitalisation des connaissances acquises sur le fonctionnement du système au cours de son adaptation. Ainsi, les résultats des différentes analyses permettraient de définir pour chaque module des plages de fonctionnement optimal en fonction, notamment, de la valeur de leur paramétrage. Il serait envisageable par la suite d'exploiter ces connaissances lors de la conception de nouveaux systèmes.

Enfin, puisque nous avons adopté le principe d'un fonctionnement séquentiel des systèmes, le relâchement de cette contrainte en vue de traiter des systèmes fonctionnant en parallèle s'avèrera tout aussi nécessaire et deviendra une source de nouveaux développements à notre méthodologie.

Cinquième partie

Annexes

A

Une méthodologie d'évaluation orientée usage des systèmes OCR commerciaux

Sommaire

A.1 Introduction

Les travaux exposés dans cette annexe se situent à la croisée de diverses problématiques abordées dans les autres parties de ce manuscrit, nommément, la définition de mesures d'évaluation, l'établissement d'un diagnostic de responsabilité ou encore la détermination de la valeur descriptive des textes vidéos.
L'objet est ici de proposer une mesure d'évaluation d'un OCR commercial (pratiquant la détection et la reconnaissance) prenant en compte dans le même temps les aspects "*algorithmiques*" classiques (écart entre la zone proposée par la détection et la zone de la vérité terrain, etc) et les aspects "*sémantiques*", liés à l'orientation usage de la mesure. Ainsi, le système est pénalisé relativement à la valeur sémantique des textes oubliés, correctement reconnus ou reconnus avec des erreurs, selon les impératifs applicatifs considérés [26]. Typiquement, un système habituellement considéré comme performant car permettant d'extraire la majorité des textes présents dans le document, pourra voir ici son comportement estimé insuffisant si les textes correctement détectés ne revêtent pas un intérêt descriptif notable.
Par ailleurs, cette mesure permet plus généralement d'évaluer un OCR commercial hors du contexte habituel dans lequel de tels systèmes sont examinés, à savoir, celui des documents entièrement *textuels*, tels que les formulaires, pages de journaux ou autre. En effet, bien que certains systèmes OCR commerciaux soient capables d'extraire et de reconnaître les textes inclus dans des images issues d'un flux audiovisuel, ceux-ci n'ont jamais été évalués dans de telles conditions d'application, la communauté scientifique considérant communément que de tels systèmes

[26] Le contexte applicatif sera limité ici au cadre de l'indexation.

ne peuvent obtenir de résultats satisfaisants sur des images d'une aussi mauvaise qualité. Pour autant, la résolution des documents vidéos augmente dans le même temps que les artefacts dus à la compression diminuent, permettant ainsi à des systèmes commerciaux d'obtenir des résultats somme toute corrects sur des images issues d'un flux audiovisuel. La question de leur évaluation dans un tel contexte d'application se trouve alors soulevée.

A.2 Méthodologie

A.2.1 Construction des vérités terrains

Les vérités terrains des textes vidéos sont construites selon la même méthodologie que celle évoquée dans le chapitre 1 de la partie 3 (la hiérarchie mot, lignes, blocs est ainsi conservée). Deux informations supplémentaires sont ici prises en compte :

- un texte peut apparaître plusieurs fois dans un même document sans pour autant être employé dans le même sens. Chaque document est ainsi découpé en segments sémantiquement homogènes et chaque texte se voit alors attribué l'identifiant du segment auquel il appartient.
- selon le même principe que dans le chapitre 1 de la partie 3, le score TFIDF des textes vidéos apparaissant simultanément dans la notice descriptive du document auquel ils appartiennent est calculé [27] :

$$TFIDF = \underbrace{qtf_t \frac{(K+1) * tf_{t,d}}{K * (1 - b + b * L_d) + tf_{t,d}}}_{TF} \underbrace{log(\frac{N}{N_t})}_{IDF} \tag{A.1}$$

où qtf_t (fixé à 1) est le nombre d'occurence du terme t dans la requête, $tf_{t,d}$ est le nombre d'occurences du terme t dans le document d, N_t est le nombre de documents dans la collection considérée contenant le terme t au moins une fois, N est le nombre total de documents dans la collection et L_d est la longueur du document d divisée par la longueur moyenne des documents de la collection. Les paramètres b et K sont fixés empiriquement à 0,86 et 1,2.

A.2.2 Evaluation

On considère ici que tout système OCR peut être appréhendé comme la séquence de trois modules : la détection, la localisation (où est situé précisément le texte dans l'image ?) et la reconnaissance [28]. L'enjeu de la première phase de l'évaluation est alors de déterminer la cause des erreurs du système.

Les textes parfaitement reconnus et les oublis (erreurs de détection) sont tout d'abord isolés. Pour les textes restant, l'origine de l'erreur est attribuée à la phase de localisation ou de

[27] Il est donc à remarquer ici que la vérité terrain *exhaustive* est filtrée : seuls les textes apparaissant conjointement dans la notice du document sont retenus.

[28] Si le système utilisé pour les expérimentations produit bien deux sorties différentes relativement à la localisation et à la reconnaissance, le module de détection ne produit pas quant à lui de sortie distincte et ne revêt donc pas de *réalité informatique* comme les deux précédents modules. A la différence des systèmes manipulés dans les autres parties de ce manuscrit, la composition en modules considérée ici est donc purement fictive.

reconnaissance. Considérant que les fausses alarmes ne sont pas un obstacle à l'objectif d'indexation (elles peuvent être facilement filtrées par un dictionnaire par exemple), celles-ci ne sont pas prises en compte. Un texte désignera par la suite soit un mot, une ligne ou un bloc. L'évaluation s'effectuera au niveau des segments sémantiques composant le document considéré.

A.2.2.1 Les textes correctement reconnus

Un même texte peut apparaître plusieurs fois dans un même segment sémantique. Dans ce cas, celui-ci sera considéré comme reconnu si au moins une de ses occurences dans le segment l'est (distance de Levenstein nulle avec la transcription de la vérité terrain).

A.2.2.2 Les textes oubliés

Un texte est considéré comme oublié si aucun membre de sa famille (mot, ligne ou bloc) n'est correctement détecté. Le critère de comparaison est basé sur les barycentres des zones : une zone est associée à une autre si son barycentre est inclus dans l'autre zone et inversement.

A.2.2.3 Les textes "non correctement reconnus"

L'erreur est imputée à la phase de reconnaissance s'il n'existe aucune localisation permettant d'obtenir un meilleur résultat de reconnaissance. Le comportement du système étant supposé homogène sur toutes les occurences d'un même texte, nous explorons le voisinage d'une unique zone : la zone (OCR_+) la plus proche de la vérité terrain au sens de la distance $d_{recouv}(GT, OCR)$:

$$d_{recouv}(GT, OCR) = \frac{A((GT \cup OCR) \setminus (GT \cap OCR))}{A(GT \cap OCR)}$$

La phase de reconnaissance est appliquée sur plusieurs zones du voisinage de OCR_+. En comparant les résultats de la reconnaissance obtenus sur celles-ci avec celui produit sur OCR_+, l'erreur est finalement imputée au module de reconnaissance ou à celui de la localisation.

A.2.2.4 L'indice global d'évaluation

L'indice global d'évaluation (cf A.2) prend en compte les informations obtenues par le précédent diagnostic (I^A_{OCR} où A fait référence à l'aspect **A**lgorithmique) ainsi que les informations relatives à l'utilité des résultats dans le cadre de l'indexation ($I^{semantic}_{OCR}$, **S**émantique).

$$\boxed{\mathbf{I_{OCR}} = \mathbf{s_{TFIDF}} \times \mathbf{I^S_{OCR}} + (\mathbf{1} - \mathbf{s_{TFIDF}}) \times \mathbf{I^A_{OCR}}}$$

$$I^A_{OCR} = \frac{N - (w_{RI}N_{RI} + w_{LI}N_{LI} + w_F N_F)}{N} \tag{A.2}$$
$$I^S_{OCR} = \frac{2 \times Précision \times Rappel}{Précision + Rappel}$$

avec

$$Précision = \frac{N^{high}_R}{N_R} \; et \; Rappel = \frac{N^{high}_R}{N^{high}_{W+F} + N^{high}_R} \tag{A.3}$$

$I_{OCR} = 1$ si $N_R = N$ et $I_{OCR} = 0$ si $N_F = N$. N_R, N_O, N_{RI} *et* N_{LI} désignent respectivement le nombre de textes reconnus, oubliés ou pour lesquels l'erreur a été attribuée à la phase de reconnaissance ou de localisation. Les coefficients w_{RI}, w_{LI} et w_O ($w_{RI} + w_{LI} + w_F = 1$) sont choisis

selon l'application finale. N_R^{high} (N_{W+O}^{high}) représentent le nombre de textes reconnus (oubliés ou "non correctement reconnus") dont le score TFIDF est élevé. s_{TFIDF}, qui représente l'écart type normalisé des scores TFIDF, permet de pondérer l'influence de I_{OCR}^S. On remarquera par ailleurs que I_{OCR}^S représente la moyenne harmonique dont il a déjà été question dans le chapitre 1 de la partie 2.

A.3 Expérimentations

La méthodologie est appliquée sur un journal télévisé (en résolution 720*576) divisé en 26 segments sémantiques (les segments sémantiques correspondent ici aux différents reportages composant le journal). La vérité terrain contient 477 éléments : 286 mots, 118 lignes et 73 blocs. 66 mots sont de type *texte de scène*. Pour le calcul des scores TFIDF une base de 58093 notices descriptives est utilisée. Le système OCR FineReader (repérage des zones de texte et reconnaissance) est appliqué pour des raisons de coût toutes les dix images. Ses résultats sont stockés dans le même format XML que celui utilisé pour les vérités terrains.
Les résultats de la phase de diagnostic sont les suivants : 58.40% des mots sont correctement reconnus (en ne considérant que les textes artificiels, ce résultats monte à 70%), 22% sont oubliés (63% des textes *de scène* le sont). Enfin, parmi les 56 mots restant, 23 erreurs sont imputées à la phase de localisation et 33 à la phase de reconnaissance.
Pour le calcul de l'indice global, les paramètres sont fixés de sorte que les erreurs de reconnaissance soient les plus pénalisées : $w_{RI} = 3 * w_{LI}$ *et* $w_F = 5 * w_{RI}$. On obtient alors $I_{OCR}^{algorithmic} = 0.82$.
En ce qui concerne l'aspect sémantique, le plus difficile est de déterminer quels sont, à partir des scores TFIDF, les textes intéressants. Le but étant de qualifier la valeur descriptive de chaque texte et non de leur ensemble, il est cette fois impossible de reposer sur le principe de comparaison de la distributions des scores TFIDF des textes vidéos avec celles des textes extraits des différents champs de la notice (cf chapitre 1 de la partie 3). Nous préférons ainsi scinder la plage de valeurs des scores TFIDF pour obtenir une vue d'ensemble des résultats (illustrée dans le tableau A.1) et laisser à l'utilisateur le choix de pénaliser plus ou moins sévèrement le système (selon l'indice *SemanticTolerance* de la plage de valeur du score TFIDF considérée).

SemanticTolerance	5	4	3	2	1
Th_{TFIDF}	1.44	4.32	7.20	10.08	12.96
$I_{OCR}^{Semantic}$	0.69	0.64	0.43	0.17	0.04
I_{OCR}	0.77	0.76	0.69	0.60	0.56

TAB. A.1 – Evolution de $I_{OCR}^{semantic}$ et I_{OCR} selon la *tolérance sémantique*

Th_{TFIDF} désigne dans le tableau A.1 le seuil utilisé pour déterminer les textes intéressants. Il apparaît dans ce tableau que les performances du système sont proportionnelles à la tolérance concernant l'importance des textes devant être correctement reconnus.

A.4 Conclusions

La méthodologie exposée dans cet annexe est orientée usage : elle donne d'une part des informations sur le mode de fonctionnement du système d'OCR grâce à un diagnostic de ses erreurs et elle permet de quantifier l'apport du système dans un cadre applicatif donné. Cette méthodologie pourrait ainsi être utilisée pour comparer plusieurs systèmes OCR relativement à

une application donnée dans l'objectif de choisir la plus adéquate aux besoins.
Il serait par ailleurs envisageable de prendre en compte les effets de post-traitements tel que celui exposé dans [Jol04], permettant d'obtenir un unique résultat de reconnaissance à partir de plusieurs résultats obtenus sur différentes occurences.
Enfin, comme cela a déjà été évoqué dans le chapitre 1 de la partie 2, cette méthodologie ne peut être étendue facilement à des systèmes dont la composition en modules s'avère plus complexe, la phase de diagnostic devenant alors particulièrement difficile. Ceci constitue ainsi la seule réelle limitation à son utilisation.

Bibliographie

[AD99] L. Agnihotri and N. Dimitrova. Text detection for video analysis. In *Workshop on Content Based Access of Image and Video Librairies, held in conjunction with CVPR*, pages 109–113, Fort Collins, CO, USA, 1999.

[AFK01] D. Argiro, K. Farrar, and S. Kubica. Cantata : the visual programming environnement for the khoros system. In *Proceedings of the IASTED International Conference (Visualization, imaging and image processing)*, pages 697–702, Marbella, Spain, 2001.

[All03] A. Allauzen. *Modélisation linguistique pour l'indexation automatique de documents audiovisuels*. PhD thesis, Laboratoire LIMSI, 2003.

[AMS+03] H. Azzag, N. Monmarché, M. Slimane, G. Venturini, and C. Guinot. A clustering algorithm based on the ants self-assembly behaviour. In W. Banzhaf, T. Christaller, P. Dittrich, J. T. Kim, and J. Ziegler, editors, *Advances in Artificial Life - Proceedings of the 7th European Conference on Artificial Life (ECAL)*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 564–571, Dortmund, Allemagne, Septembre 2003. Springer Verlag Berlin, Heidelberg.

[AR91] Y. Aloimonos and A. Rosenfeld. Computer vision. *Science*, 253 :1249–1254, 1991.

[AS94] C. Ahlberg and B. Schneiderman. Visual information seeking using the filmfinder. In *CHI '94 : Conference companion on Human factors in computing systems*, pages 433–434, Boston, MA, USA, 1994. New York : ACM Press.

[Bac99] B. Bachimont. Bibliothèques numériques audiovisuelles : des enjeux scientifiques et techniques. *Document Numérique*, 2(3-4) :219–242, 1999.

[Bac01] B. Bachimont. Audio-visual indexing and automatic analysis how to bridge the gap. In *MMCBIR 2001 - Indexation et Recherche par le Contenu dans les Documents Multimedia*, INRIA, Rocquencourt, 2001.

[BBTR04] M. Bertozi, A. Broggi, A. Tribaldi, and M. Del Rose. A tool for vision based pedestrian detection performance evaluation. In *IEEE Intelligent Vehicle Symposium*, Parma, Italy, 2004.

[BER03] J. Black, T. Ellis, and P. Rosin. A novel method for video tracking performance evaluation. In *Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 125–132, Nice, France, 2003.

[BG97] R. Bakeman and J.M. Gottman. *Observing interaction : An introduction to sequential analysis (2nd ed.)*, chapter Assessing observer agreement. New York : Cambridge University Press, 1997.

[BM01] S. Bandyopadhyay and U. Maulik. Nonparametric genetic clustering : comparison of validity indices. *IEEE Transactions on Systems, Man and Cybernetics, Part C : Applications and Reviews*, 31(1), 2001.

[BM02] S. Bandyopadhyay and U. Maulik. Genetic clustering for automatic evolution of clusters and application to image classification. *Pattern Recognition*, 35(6) :1197–1208, 2002.

[BST+05] L.M. Brown, A.W. Senior, Y-L. Tian, J. Connell, A. Hampapur, C-F. Shu, H. Merkl, and M. Lu. Performance evaluation of surveillance systems under varying conditions. In *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, Breckenridge, CO, USA, 2005.

[BSW99] D. Bikel, R. Schwartz, and R.M. Weischedel. An algorithm that learns what's in a name. *Machine Learning, Special Issue on NL Learning*, 34 :211–231, 1999.

[Bur05] N. Burrus. Détection statistique de segments significatifs sur rétine programmable, 2005. Rapport de DEA. Paris : ENSTA. 20 p.

[CBT01] D. Chen, H. Bourlard, and J-P. Thiran. Text identification in complex background using svm. In *International Conference on Computer Vision and Pattern Recognition*, pages 621–626, Kauai Marriott, HI, USA, 2001.

[CCR05] M. Caillet, J. Carrive, and C. Roisin. Description des documents audiovisuels : s'affranchir des limitations de mpeg-7 fdl, vers un langage de description par objets, extensible et modulaire. In *INFORSID 2005 Atelier METSI*, Grenoble, France, 2005.

[CEPR99] R. Clouard, A. Elmoataz, C. Porquet, and M. Revenu. Borg : a knowledge-based system for automatic generation of image processing programs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(2) :128–144, 1999.

[CGdLM03] A. Casillas, M. T. González de Lena, and R Martínez. Document clustering into an unknown number of clusters using a genetic algorithm. *Text, Speech and Dialogue TSD 2003. Lecture Notes in Artificial Intelligence, series of Lecture Notes in Computer Science*, pages 43–49, 2003.

[Cha05] J. Charlet. L'ingénierie des connaissances, une science de gestion ? In R. Teulier and P. Lorino, editors, *Entre la connaissance et l'organisation, l'activité collective*, chapter 11. La découverte, 2005.

[Clo04] R. Clouard. Une méthode de développement d'applications de traitement d'images. *Traitement du Signal*, 21(4) :277–293, 2004.

[COB04] D. Chen, J-M. Odobez, and H. Bourlard. Text detection and recognition in images and video frames. *Pattern Recognition*, (37) :595–608, 2004.

[CPER95] R. Clouard, C. Porquet, A. Elmoataz, and M. Revenu. Why building knowledge-based image segmentation is so difficult. In *International Workshop on Knowledge-based systems for the reuse of program librairies*, Sophia Antipolis, France, 1995.

[CSFN02] P.Y. Chen, R. Srinivasan, G. Fedosejevs, and B. Narasimhan. An automated cloud detection method for daily noaa-14 avhrr data for texas. *International Journal of Remote Sensing*, 23(15) :2939–2950, 2002.

[CVRD06] C. Clavel, I. Vasilescu, G. Richard, and L. Devillers. Du corpus émotionnel au système de détection : le point de vue applicatif de la surveillance dans les lieux publics. *Revue d'Intelligence Artificielle*, 2006. A paraître.

[CWK03] Y. Chen, J.Z. Wang, and R. Krovetz. Content-based image retrieval by clustering. In *MIR '03 : Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 193–200, New York, NY, USA, 2003. ACM Press.

[DADB00] N. Dimitrova, L. Agnihotri, C. Dorai, and R. Bolle. Mpeg-7 videotext description scheme for surimposed text in images and video. *Signal Processing : Image Communication*, 16 :137–155, 2000.

[DB79] D.L. Davies and D.W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(4) :224–227, 1979.

[DBB00] B.A. Draper, J. Bins, and K. Baek. Adore : Adaptative object recognition. *VIDERE*, 1(4) :86–99, 2000.

[DD98] P. Dalle and P. Dejean. Planification en traitement d'image : approche basée sur les données. In *Congrès RFIA*, pages 75–84, Clermont-Ferrand, France, 1998.

[DH96] B. Draper and A.R. Hanson. Knowledge-directed vision : control, learning and integration. *Proceedings of IEEE*, 84(11) :1625–1681, 1996.

[DHS01] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification, second edition.* New York : Wiley And Sons, 2001. 654 p.

[DM00] D. Doermann and D. Mihalcik. Tools and techniques for video performance evaluation. In *Proceedings of the ICPR 2000*, volume 4, pages 167–170, Barcelona, Spain, 2000.

[DM02] J.G. Digalakis and K.G. Margaritis. An experimental study of benchmarking for genetic algorithms. *International Journal of Computer Mathematics*, 79(4) :403–416, 2002.

[DMEM98] Nevenka Dimitrova, Thomas McGee, Herman Elenbaas, and Jacquelyn Martino. Video content management in consumer devices. *IEEE Transactions on Knowledge and Data Engineering*, 10(6) :988–995, 1998.

[DMM01] A. Desolneux, L. Moisan, and J-M. Morel. Edge detection by helmholtz principle. *Journal of Mathematical Imaging and Vision*, 14(3) :271–284, 2001.

[Dub87] R.C. Dubes. How many clusters are best ? - an experiment. *Pattern Recognition*, 20(6) :645–663, 1987.

[Dun74] J. Dunn. Well separated clusters and optimal fuzzy partitions. *J. Cybernetics*, 4 :95–104, 1974.

[FCPR98] V. Ficet-Cauchard, C. Porquet, and M. Revenu. An interactive case-based reasoning system for the development of image processing applications. In *European Workshop on Case Base Reasoning (EWCBR)*, pages 437–447, Dublin, Irlande, 1998.

[Fis36] R.A. Fisher. The use of multiple mesurements in taxonomic problems. *Annals of Eugenics*, 7(2) :179–188, 1936.

[FK96] H. Frigui and R. Krishnapuram. A robust algorithm for automatic extraction of an unknown number of clusters from noisy data. *Pattern Recognition Letters*, 17(12) :1223–1232, 1996.

[FMC99] M.A. Fard, W. Mahdi, and L. Chen. Exterior and interior shot classification for automatic content-based video indexing. *Multimedia Storage and Archiving Systems IV*, 3846(1) :46–55, 1999.

[Fru95] T. Fruchterman. Dafs : A standard for document and image understanding. In *Proceedings of the Symposium on Document Image Understanding Technology*, pages 94–100, Bowie, MD, USA, 1995.

[GA00] C. Garcia and X. Apostolidis. Text detection and segmentation in complex color images. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages IV–2326–2329, Istanbul, Turquie, 2000.

[Gao03] Y. Gao. Population size and sampling complexity in genetic algorithms. In *Proceedings of the Bird of a Feather Workshops(GECCO2003)—Learning, Adaptation, and Approximation in Evolutionary Computation*, pages 178–181, Chicago, IL, USA, 2003.

[Gar00] C. Garbay. *Les systèmes de vision*, chapter Architectures logicielles et contrôle dans les systèmes de vision, pages 197–251. Paris : Hermes, 2000.

[Gom04] J. Gomez. Self adaptation of operator rates in evolutionnary algorithms. *Lecture Notes in Computer Science*, 3102 :1162–1173, 2004.

[Gre03] W.A. Greene. Unsupervised hierarchical clustering via a genetic algorithm. In *Proceedings of the Congress on Evolutionary Computation*, Camberra, Australie, 2003. IEEE Press.

[Gru93] T.R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5 :199220, 1993.

[Hat85] J-P. Haton. Intelligence artificielle en compréhension automatique de la parole : état des recherches et comparaison avec la vision par ordinateur. *TSI*, 4(3) :265–287, 1985.

[HCWZ01] X-S. Hua, X-R. Chen, L. Wenyin, and H-J. Zhang. Automatic location of text in video frames. In *3rd International Workshop on Multimedia Information Retrieval, in Conjonction with ACM Multimedia*, Ottawa, Canada, 2001.

[Hol75] J. Holland. Adaptation in natural and artificial systems, Ann Arbor : The University of Michigan, 1975. 183 p.

[HT03] C. Hudelot and M. Thonnat. A cognitive vision platform for automatic recognition of natural complex objects. In IEEE, editor, *International Conference on Tools with Artificial Intelligence, ICTAI, Sacramento*, 2003.

[HWJ98] L. Hong, Y. Wan, and A.K. Jain. Fingerprint image enhancement : Algorithm and performance evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8) :777–789, 1998.

[HWZ04a] X-S. Hua, L. Wenyin, and H-J. Zhang. An automatic performance evaluation protocol for video text detection algorithms. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(4) :498–507, 2004.

[HWZ04b] X-S. Hua, L. Wenyin, and H-J. Zhang. An automatic performance evaluation protocol for video text detection algorithms. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(4) :498–507, 2004.

[HYZ02] X-S. Hua, P. Yin, and H-J. Zhang. Efficient video text recognition using multiple frame integration. In *International Conference on Image Processing (ICIP)*, pages II–397–II–400, Rochester, NY, USA, 2002.

[IOO91] S. Impedovo, L. Ottaviano, and S. Occhinegro. Optical character recognition-a survey. *International Journal on Pattern Recognition and Artificial Intelligence*, 5(1-2) :1–24, 1991.

[JD96] Ph. Jean and P. Dalle. Un langage de description de concepts pour la formulation d'objectifs d'analyse. In *ORASIS*, Clermont-Ferrand, France, 1996.

[JFB03] A. Joly, C Frélicot, and O. Buisson. Robust content-based video copy identification in a large reference database. In *Lecture Notes in Computer Science*, volume 2728, pages 414–424. Springer, 2003.

[Jol87] J-M. Jolion. *Méthodologie de conception de systèmes d'analyse d'images : application à la microscopie électronique*. PhD thesis, Lyon : Institut National des Sciences Appliquées de Lyon, 1987. 530 p.

[Jol00] J-M. Jolion. *Les systèmes de vision*, chapter Sur la méthodologie de conception des systèmes de vision, pages 95–129. Paris : Hermes, 2000.

[Jol04] J.M. Jolion. The deviation of a set strings. *Pattern Analysis and Applications*, 6(3) :224–231, 2004.

[Jun01] K. Jung. Neural network-based text location in color images. *Pattern Recognition Letters*, (22) :1503–1515, 2001.

[JWR98] K. Sparck Jones, S. Walker, and S.E. Robertson. A probabilistic model of information retrieval : development and status. Technical Report Technical Report 446, University of Cambridge Computer Laboratory, 1998.

[KGCM04] J. Kang, K. Gajera, I. Cohen, and G. Medioni. Detection and tracking of moving objects from overlapping eo and ir sensors. In *Joint IEEE International Workshop on Object Tracking and Classification Beyond the Visible Spectrum (OTCBVS'04)*, Washington, D.C., USA, 2004.

[KJPhK01] K.I. Kim, K. Jung, S.H. Park, and h.J. Kim. Support vector machine-based text detection in digital video. *Pattern Recognition*, (34) :527–529, 2001.

[KR94] K. Konstandinides and J.R. Rasure. The khoros software development environment for image and signal processing. *IEEE Transactions on Image Processing*, 3(3) :243–252, 1994.

[LBF05] X. Liu, K.W. Bowyer, and P.J. Flynn. Experimental evaluation of iris recognition. In *IEEE Workshop on Face Recognition Grand Challenge Experiments*, page 158, San Diego, CA, USA, 2005.

[LD99] H. Li and D. Doermann. Text enhancement in digital video using multiple frame integration. In *ACM Multimedia (1)*, pages 19–22, 1999.

[LD01] H. Li and D. Doermann. Text quality estimation in video. In *Proceedings of SPIE, Document Recognition and Retrieval IX*, volume 4670, pages 232–243, 2001.

[LDK00] H. Li, D. Doermann, and O. Kia. Automatic text detection and tracking in digital videos. *IEEE Transactions on Image Processing - Special Issue on Image and Video Processing for Digital Libraries*, 9(1) :147–156, 2000.

[Lem03] A. Lemieux. Système d'identification de personnes par vision numérique, 2003. Mémoire de Maîtrise, Québec : Faculté des Etudes Supérieures, Université Laval, 163 p.

[LF94] E.D. Lumer and B. Faieta. Diversity and adaptation in populations of clustering ants. In *SAB94 : Proceedings of the third international conference on Simulation of adaptive behavior : from animals to animats 3*, pages 501–508, Cambridge, MA, USA, 1994. MIT Press.

[LJK03] C.W. Lee, K. Jung, and H.J. Kim. Automatic text detection and removal in video sequences. *Pattern Recognition Letters*, (24) :2607–2623, 2003.

[LK03] C. H. Lee and T. Kanungo. The architecture of trueviz : a groundtruth/metadata editing and visualizing toolkit. *Pattern Recognition*, 36(3) :811–825, 2003.

[LPH97] J. Liang, I.T. Phillips, and R.M. Haralick. Performance evaluation of document layout analysis algorithms on the uw data set. In *Document Recognition IV, Proceedings of the SPIE*, pages 149–160, San Jose, CA, USA, 1997.

[LVJ05a] R. Landais, L. Vinet, and J-M. Jolion. Analyse d'un système de détection d'objets dans un flux vidéo en vue de son adaptation. In *CORESA 2005*, Rennes, France, 2005.

[LVJ05b] R. Landais, L. Vinet, and J-M. Jolion. Evaluation of commercial ocr : A new goal directed methodology for video documents. In *ICAPR (1)*, pages 674–683, 2005.

[LW02] R. Lienhart and A. Wernike. Localizing and segmenting text in images, videos and web pages. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(4) :256–268, 2002.

[LWVJ03] R. Landais, C. Wolf, L. Vinet, and J-M. Jolion. Utilisation de connaissances a priori pour le paramétrage d'un algorithme de détection de textes dans les documents audiovisuels. application à un corpus de journaux télévisés. In *Congrès RFIA*, Toulouse, France, 2003.

[Mar82] D. Marr. *Vision.* San Fransisco : Freeman, 1982. 397 p.

[MJR90] P. Meer, J-M. Jolion, and A. Rosenfeld. A fast parallel algorithm for blind estimation of noise variance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12 :216–223, 1990.

[MK95] J. Matas and J. Kittler. Spatial and feature space clustering : applications in image analysis. In *International Conference on Computer Analysis of Images and Patterns*, pages 162–173, Prague, République Tchèque, 1995.

[MM99] S. Messelodi and C.M. Modena. Automatic identification and skew estimation of text lines in real scene images. *Pattern Recognition*, 32(5) :791–810, 1999.

[MMP+02] V. Y. Mariano, J. Min, J-H. Park, R. Kasturi, D. Mihalcik, H. Li, and D. Doermann. Performance evaluation of object detection algorithms. In *International Conference on Pattern Recognition*, pages 965–969, Québec, Canada, 2002.

[Moi03] L. Moisan. Modèles continus, numériques et statistiques pour l'analyse d'images, HDR, Paris : Université Paris-Sud, Centre d'Orsay, 2003. 60 p.

[MTB04] N. Maillot, M. Thonnat, and A. Boucher. Towards ontology based cognitive vision. *Machine Vision and Applications*, 16(1) :33–40, 2004.

[MTH04] N. Maillot, M. Thonnat, and C. Hudelot. Ontology based object learning and recognition : Application to image retrieval. In *Proceedings of 16th IEEE International Conference on Tools For Artificial Intelligence*, Boca Raton, USA, 2004. IEEE Computer Society Press.

[MVvdEvH95] S. Moisan, R. Vincent, J. van den Elst, and F. van Harmelen. Towards an intelligent failure handling mechanism in program supervision. In *1st International Workshop on Knowledge Based systems for the (re)Use of Program Libraries*, pages 109–118, Sophia Antipolis, France, Nov 1995.

[NHBM98] D.J. Newman, S. Hettich, C.L. Blake, and C.J. Merz. UCI repository of machine learning databases, Irvine CA : University of California, Department of Information and Computer Sciences, 1998.

[Nib85] W. Niblack. *An introduction to digital image processing.* Birkeroed, Danemark : Strandberg Publishing Company, 1985. 215 p.

[PBC97] V. Parameswaran, P. Burlina, and R. Chellappa. Performance analysis and learning approaches for vehicle detection and counting in aerial images. In *ICASSP*, pages 2753–2756, Munich, Allemagne, 1997.

[PBR98] C. Papin, P. Bouthemy, and G. Rochard. Detection of low clouds in meteosat images based on a contextual spatio-temporal labeling approach. In *IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 561–565, Chicago, IL, USA, 1998.

[PCMT01] A. Prati, R. Cucchiara, I. Mikic, and M.M. Trivedi. Analysis and detection of shadows in video streams : A comparative evaluation. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages II–571–II–576, Kauai Marriott, HI, USA, 2001.

[PLE01] S. Pfeiffer, R. Lienhart, and W. Effelsberg. Scene determination based on video and audio features. *Multimedia Tools and Applications*, 15 :59–81, 2001.

[PMK+04] R. Prasad, S. Matsoukas, C.-L. Kao, J. Ma, D.-X. Xu, T. Colthurst, G. Thattai, O. Kimball, R. Schwartz, J.-L. Gauvain, L. Lamel, H. Schwenk, G. Adda, and F. Lefevre. The 2004 BBN/LIMSI 20xRT English Conversational Telephone Speech System. In *Proc. DARPA RT04*, Palisades NY, November 2004.

[PMRR00] P.J. Philips, H. Moon, S.A. Rizvi, and P.J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10) :1090–1104, 2000.

[Pol03] J.P. Poli. Segmentation des journaux télévisés en plateaux/reportages par comparaison d'images, 2003. Rapport de DEA, Paris : DEA IARFA, Université Paris 6.

[PP00] P. Pu and Z. Pecenovic. Dynamic overview techniques for image retrieval. In *VisSym'00,Second Joint Eurographics-IEEE TCVG Symposium on Visualization*, 2000.

[PRTM04] V. Popovici, Y. Rodriguez, J.-P. Thiran, and S. Marcel. On performance evaluation of face detection and localization algorithms. In *17th International Conference on Pattern Recognition*, volume 1, pages 313–317, Cambridge, Angleterre, 2004.

[RCR05] A. Renouf, R. Clouard, and M. Revenu. Un modèle de formulation d'applications de traitement d'images. In *ORASIS*, Fournol, France, 2005. 10 p.

[Rou87] P.J. Rousseeuw. Sihouettes : a graphical aid to the interpretation and validation of cluster analysis. *J. Comp App. Math.*, 20 :53–65, 1987.

[RPM02] S. Rizvi, P.J. Phillips, and H. Moon. The FERET verification testing protocol for face recognition algorithms. *Image and Vision Computing - Special Issue on Face and Gesture Recognition*, page 48, 2002.

[Sav02] A. Savakis. A computationally efficient approach to indoor/outdoor scene classification. In *ICPR '02 : Proceedings of the 16 th International Conference on*

Pattern Recognition (ICPR'02) Volume 4, pages 40–46, Washington, DC, USA, 2002. IEEE Computer Society.

[SBK99] K. Sobottka, H. Bunke, and H. Kronenberg. Identification of text on colored book and journal covers. In *International Conference on Document Analysis and Recognition ICDAR*, pages 57–62, Bangalore, Inde, 1999.

[SBM04] Z. Sun, G. Bebis, and R. Miller. Object detection using feature subset selection. *Pattern Recognition*, 37 :2165–2176, 2004.

[SBM05] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection using evolutionary gabor filter optimization. *IEEE Transactions on Intelligent Transportation Systems*, 6(2) :125–137, 2005.

[Sch97] B. Schiele. *Reconnaissance d'objets utilisant des histogrammes multidimensionnels de champs réceptifs.* PhD thesis, Grenoble : Laboratoire GRAVIR-IMAG, INPG, 1997. 161 p.

[SD98] C. Shim and C. Dorai. Interframe analysis for improved text extraction from video. Technical report, IBM T.J. Watson Research Center, Yorktown Heights, 1998. Working Document.

[SD99] J-C. Shim and C. Dorai. A generalized region labeling algorithm for image coding, restoration, and segmentation. In *IEEE International Conference on Image Processing*, volume 1, pages 46–50, 1999.

[SDB98] J-C. Shim, C. Dorai, and R. Bolle. Automatic text extraction from video for content-based annotation and retrieval. In *International Conference on Pattern Recognition*, pages 618–620, Brisbane, Australie, 1998.

[SGPOB05] K. Smith, D. Gatica-Perez, J-M. Odobez, and S. Ba. Evaluating multi-object tracking. In *Workshop On Empirical Evaluation Methods in Computer Vision (EEMCV)*, San Diego, CA, USA, 2005.

[Shi03] H. Shiaw. New interaction techniques for the digital library : 3d focus+context interactive visualization. Technical Report 2003-7, Department of Computer Science, Tufts University, 2003.

[Shu99] J.A. Shufelt. Performance evaluation and analysis of monocular building extraction from aerial imagery. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4) :311–326, 1999.

[SKH+99] T. Sato, T. Kanade, E.K. Hughes, M.A. Smith, and S. Satoh. Video ocr : Indexing digital news librairies by recognition of surimposed captions. *ACM Multimedia Systems Special Issue on Video Libraries*, 7(5) :385–395, 1999.

[SLB+05] S. Steidl, M. Levit, A. Batliner, E. Nöth, and H. Niemann. "of all things the measure is man" automatic classification of emotions and inter-labeler consistency. In *ICASSP : International Conference on Acoustics, Speech, and Signal Processing*, pages 317–320, 2005.

[SMV+98] C. Shekhar, S. Moisan, R. Vincent, P. Burlina, and R. Chellappa. Knowledge-based control of vision systems. *Image and Vision Computing*, 17 :667–683, 1998.

[SNK97] S. Satoh, Y. Nakamura, and T. Kanade. Name-it : Naming and detecting faces in video by the integration of image and natural language processing. In *Proceedings of the 1997 International Joint Conference on Artificial Intelligence (IJCAI '97)*, pages 1488–1493, Nagoya, Japon, 1997.

[SP96] K. Sobottka and I. Pitas. Extraction of facial regions and features using color and shape information. In *International Conference on Pattern Recognition (ICIP)*, pages 421–425, Vienne, Autriche, 1996.

[SSHP97] J. Sauvola, T. Seppänen, S. Haapakoski, and M. Pietikaïnen. Adaptative document binarization. In *International Conference on Document Analysis and Recognition*, volume 1, pages 147–152, 1997.

[Sum01] K. Summers. Ocr accuracy of three systems on english and russian documents of highly varying quality. In *SDIUT 2001*, Columbia, Maryland ,USA, 2001.

[SWS+00] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12) :1349–1380, 2000.

[TCvdE94] M. Thonnat, V. Clement, and J. van den Elst. Supervision of perception tasks for autonomous systems : the OCAPI approach. *Journal of Information Science and Technology*, 3(2) :140–163, 1994.

[TDR04] S. Treuillet, D. Driouchi, and P. Ribereau. Ajustement des paramètres d'une chaîne de traitements d'images par un plan d'expérience factoriel fractionnaire 2^{k-p}. *Traitement du Signal*, 21(2) :141–156, 2004.

[TGLZ02] X. Tang, X. Gao, J. Liu, and H. Zhang. A spatial-temporal approach for video caption detection and recognition. *IEEE Transactions on Neural Networks*, 13(4) :961–971, 2002.

[UG96] M. Uschold and M. Grüninger. Ontologies : principles, methods, and applications. *Knowledge Engineering Review*, 11(2) :93–155, 1996.

[Vap98] V. Vapnik. *Statistical Learning theory.* New York : Wiley, 1998.

[VFJZ01] A. Vailaya, M. Figueiredo, A.K. Jain, and H. J. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1) :117–130, 2001.

[VJ04] P. Viola and M.J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(4) :137–154, 2004.

[vR79] C.J. van Rijsbergen. *Information Retrieval second edition.* Londres : Butterworths, 1979. 208 p.

[WC03] E.K. Wong and M. Chen. A new robust algorithm for video text extraction. *Pattern Recognition*, 36(6) :1397–1406, 2003.

[WCCS04] Y. Wu, E.Y. Chang, K.C.C. Chang, and J.R. Smith. Optimal multimodal fusion for multimedia data analysis. In *MULTIMEDIA '04 : Proceedings of the 12th annual ACM international conference on Multimedia*, pages 572–579, New York, NY, USA, 2004. New York : ACM Press.

[WD02] C. Wolf and D. Doermann. Binarization of low quality text using a markov random field model. In *Proceedings of the International Conference on Pattern Recognition*, volume 3, pages 160–163, 2002.

[WF74] R.A. Wagner and M.J. Fisher. The string to string correction problem. *Journal of Assoc. Comp. Mach.*, 21(1) :168–173, 1974.

[Whi94] D. Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4 :65–85, 1994.

[Whi01] D. Whitley. An overview of evolutionary algorithms. *Journal of Information and Software Technology*, 43 :817–831, 2001.

[WK03] K. Wang and J.A. Kangas. Character location in scene images from digital camera. *Pattern Recognition*, 36 :2287–2299, 2003.

[WLJ05] C. Wolf, R. Landais, and J.M. Jolion. *Trends and Advances in Content-Based Image and Video Retrieval*, chapter Detection of Artificial Text for Semantic Indexing. Lecture Notes in Computer Science. Springer Verlag, 2005. (to appear).

[WM98] V. Wu and R. Manmatha. Document image clean-up and binarization. In *Proceedings of SPIE*, pages 263–273, 1998.

[WMR99] V. Wu, R. Manmatha, and E.M. Riseman. Textfinder : An automatic system to detect and recognize text in images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(11) :1224–1229, 1999.

[Wol03] C. Wolf. *Détection de textes dans les images issues d'un flux vidéo pour l'indexation sémantique*. PhD thesis, Lyon : INSA de Lyon, 2003. 205 p.

[WP03] J.R Wang and N Parameswaran. Survey of sports video analysis : Research issues and applications. In *Pan-Sydney Area Workshop on Visual Information Processing (VIP2003)*, Sydney, Australie, 2003.

[WZ04] H. Wu and Q. Zheng. Self-evaluation for video tracking systems. In *Army Science Conference*, Orlando, FL, USA, 2004.

[XHC+01] J. Xi, X-S. Hua, X-R. Chen, L. Wenyin, and H-J. Zhang. A video text detection and recognition system. In *Proceedings of IEEE International Conference on Multimedia and Expo*, pages 873–876, Tokyo, Japon, 2001.

[YKA02] M-H. Yang, D.J. Kriegman, and N. Ahuja. Detecting faces in images : a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1) :34–58, 2002.

[YP03] Y. Yitzhaky and E. Peli. A method for objective edge detection evaluation and detector parameter selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8) :1027–1033, 2003.

[YV98] B. Yanikoglu and L. Vincent. Pink panther : A complete environment for ground-truthing and benchmarking document page segmentation. *Pattern Recognition*, 31(20) :1191–1204, 1998.

[ZCPR03] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition : A literature survey. *ACM Comput. Surv.*, 35(4) :399–458, 2003.

[ZKJ95] Y. Zhong, K. Karu, and A.K. Jain. Locating text in complex color images. *Pattern Recognition*, 28(10) :1523–1535, 1995.

[ZS02] G. Zhou and J. Su. Named entity recognition using an hmm-base chunk tagger. In *Proceedings of the 40th annual meeting of the Association of Computional Linguistics*, pages 473–480, Philadelphie, PA, USA, 2002.