N^o d'ordre HDR 2012 017 Année 2012

Mémoire

Modélisation globalement cohérente d'interactions complexes avec prise en compte de critères géométriques

Présentée devant
L'Institut National des Sciences Appliquées de Lyon

Pour obtenir
le diplôme d'Habilitation à Diriger des Recherches

Ecole Doctorale Informatique et Mathematiques

Par

Christian Wolf

Soutenue le 10 décembre 2012
devant la Commission d'Examen :

Rapporteur	Patrick Bouthemy	DR, INRIA Rennes
Rapporteur	François Bremond	DR, INRIA Sophia Antipolis
Rapporteur	Wojciech Pieczynski,	PU, Télécom SudParis
Examinateur	Vincent Charvillat	PU, Université de Toulouse
Examinateur	Bülent Sankur	Prof., Bogacizi University, Turquie
Examinateur	Atilla Baskurt	PU, INSA-Lyon
Examinateur	Florent Dupont	PU, Université Lyon 1
Examinateur	Jean-Michel Jolion	PU, INSA-Lyon

Remerciements

Je remercie Christine Solnon, Eric Lombardi, Vincent Vidal, Guillaume Lavoué, Guillaume Beslon, Emmanuel Dellandréa et Christophe Garcia pour la relecture des différents chapitres.

Je remercie Jean-Michel Jolion pour son soutien dans une période difficile.

Je remercie Atilla Baskurt, Florent Dupont et Bülent Sankur pour les collaborations fructueuses.

Je remercie ma famille pour son soutien et pour la compréhension lors des nos vacances difficiles, passées devant le PC, et je remercie mon fils Noah pour son sourire et son amour inconditionnel.

Résumé

Les recherches présentées ici traitent d'analyse d'images, de vidéos et de maillages. L'idée directrice est la modélisation d'interactions complexes entre plusieurs variables, le plus souvent réalisée à l'aide de modèles graphiques, généralement probabilistes ; la modélisation globalement cohérente d'un problème ; la résolution de problèmes complexes par minimisation de fonctions d'énergie globales ; les modèles structurés et semi-structurés : graphes, chaînes, arbres etc.

Ces travaux peuvent être globalement regroupés en quatre thèmes applicatifs :

(i) Segmentation d'images et de vidéos - les défis de cette thématique résident dans la modélisation de contenus complexes et de dégradations complexes tout en permettant une inférence efficace.

(ii) Détection et reconnaissance d'objets - ces travaux se basent essentiellement sur les modèles structurés et semi-structurés. Le verrou scientifique majeur est l'augmentation du pouvoir de discrimination d'un modèle, tout en gardant, ou en augmentant, l'invariance vis-à-vis de transformations diverses comme les changements d'échelle, les rotations, les mouvements articulés, les changements d'éclairage etc. L'inférence efficace reste un souci.

(iii) Reconnaissance d'actions - une partie de ces travaux est liée aux travaux sur la reconnaissance d'objets de par leurs contributions théoriques sur les modèles structurés et semi-structurés. Les contributions les plus notables concernent la modélisation d'activités humaines par graphes.

(iv) Analyse de maillages - l'objectif de cette thématique est la conception de modèles de graphiques pour les maillages surfaciques en vue de leur analyse, segmentation et filtrage. Dans un contexte de modélisation globalement cohérente, la difficulté principale provient de la structure très irrégulière d'un maillage.

Table des matières

Chapitre 1

CV étendu

Ce chapitre présentera d'abord quelques informations succinctes de nature biographique résumant le dossier. Les sections suivantes donneront le contexte de nos travaux de recherche, les projets auxquels ils étaient liés, les encadrements, les activités d'enseignement et d'administration.

1.1 Résumé

Biographie

Christian Wolf
39 ans, marié, 1 enfant
Nationalité Autrichienne

christian.wolf@liris.cnrs.fr
http://liris.cnrs.fr/christian.wolf

Maître de Conférences
Université de Lyon, INSA-Lyon

Unité d'attachement de recherche :
LIRIS UMR CNRS 5205
Equipe Imagine — Extraction de Caractéristiques et Identification
Equipe M2Disco — Modèles Multirésolution, Discrets et Combinatoires

Unité d'attachement d'enseignement : INSA-Lyon, Département premier cycle (50%)
INSA-Lyon, Département Informatique (50%)

PES (Prime d'excellence scientifique) depuis 2009.

Parcours

2005 - aujourd'hui	**MCF**, INSA de Lyon, LIRIS
2004 - 2005	**MCF**, Université Louis Pasteur de Strasbourg, ENSPS, LSIIT
2003 - 2004	**ATER**, INSA de Lyon, LIRIS
2003	Thèse de doctorat en informatique de l'INSA de Lyon, LIRIS : *Détection de textes dans des images issues d'un flux vidéo pour l'indexation sémantique* Prof. Catherine Berrut, Université J. Fourier de Grenoble (président) Prof. Annick Montanvert, Université P. Mendès-France de Grenoble (rapporteur) Prof. Marinette Revenu, Université de Caen (rapporteur) Prof. Jean-Michel Jolion, INSA de Lyon (directeur) Dr. David Doermann, University of Maryland (examinateur) Dr. Christophe Laurent, France Télécom R&D (examinateur)
2000	*Diplom-Ingenieur der Informatik* de *Vienna University of Technology*, Autriche

Mobilité

Bac+5 à *Vienne, Autriche* ; thèse à *Lyon* ; recrutement MCF à *Strasbourg* ; mutation à *Lyon*.
Chercheur invité à l'*Université de Maryland, USA* : 2×3 mois en 2001 et 2002.

1.2 Travaux de recherche : résumé et contexte

Nos travaux de recherche sont décrits en détail dans la deuxième partie de ce mémoire (chapitres 2 à 7). Nous renvoyons le lecteur au chapitre 2 pour une introduction succincte et pédagogique. Ici, nous nous contenterons d'esquisser très brièvement les sujets qui nous ont intéressé afin de pouvoir les lier à leur contexte, aux projets et aux encadrements.

Nos activités de recherche durant les dernières années se sont orientées vers l'analyse d'images, de vidéos et de maillages en vue de l'extraction d'informations de haut niveau ; la reconnaissance d'« objets » dans un sens très large : objets 2D, parties d'objets 2D et 3D, activités humaines, etc. Les applications couvrent la détection et la reconnaissance d'objets dans les images et dans les séquences vidéo, la détection et la reconnaissance d'activités dans les vidéos, la segmentation, l'analyse et la restauration d'images et de modèles 3D, l'indexation d'images et de séquences vidéos par le contenu etc.

L'idée directrice et le fil rouge unifiant ces thématiques sont la modélisation (souvent probabiliste) d'interactions complexes entre plusieurs variables, le plus souvent réalisée à l'aide de modèles graphiques probabilistes, tels que les champs de Markov, les réseaux Bayesiens et les chaines de Markov cachées ; la modélisation globalement cohérente d'un problème ; la résolution de problèmes complexes par minimisation de fonctions d'objectifs (fonctions d'énergie) globales ; les modèles structurés et semi-structurés : graphes, chaînes, arbres etc. Une introduction à ce genre de modèles est donnée dans le chapitre 3.

La figure 1.1 montre un graphe liant, par arêtes noires, les différents travaux majeurs, regroupés de deux manières : (i) les principaux thèmes applicatifs auxquels nous nous sommes intéressé sont

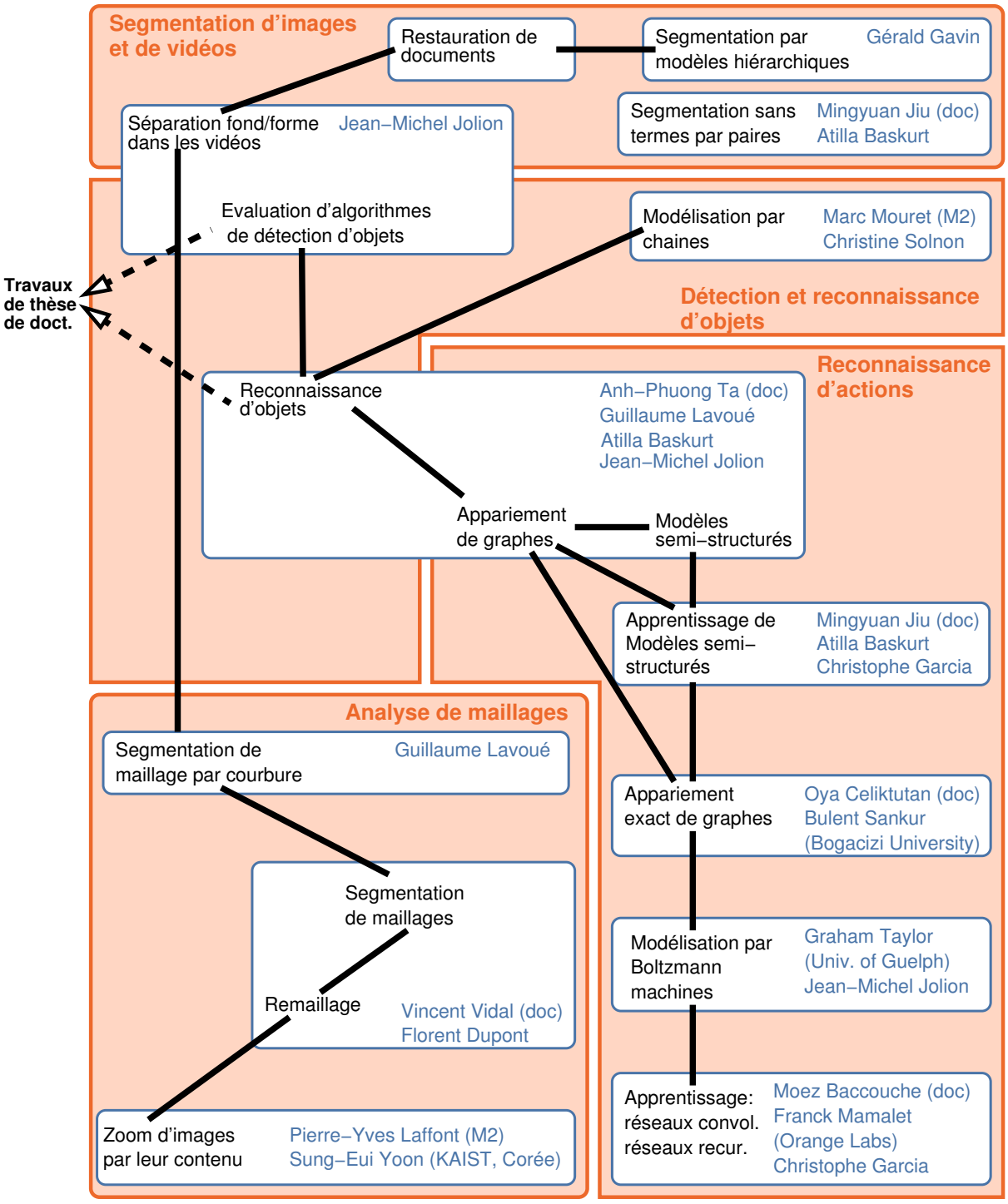


Figure 1.1 – Un graphe reliant les travaux de recherche les plus importants. Un sommet peut correspondre à plusieurs travaux et publications. Les boîtes oranges indiquent les thèmes applicatifs, les boîtes blanches les collaborations.

indiqués par des rectangles oranges ; (ii) les collaborations desquelles ils sont issus sont indiquées par des rectangles bleus. Nos travaux peuvent être globalement regroupés en cinq thèmes applicatifs :

Segmentation d'images et de vidéos il s'agit principalement de travaux personnels ; d'autres travaux ont été réalisés avec Gerald Gavin du laboratoire Eric et avec Jean-Michel Jolion du Liris. Les défis de cette thématique résident dans la modélisation de contenus complexes et de dégradations complexes tout en permettant l'inférence efficace. Dans un cadre Bayesien et dans un contexte de modélisation graphique probabiliste, nos contributions ont permis d'avancer l'état de l'art dans le domaine. Parmi nos travaux nous pouvons mentionner les modèles destinés à la restauration basés sur les double champs de Markov [3] ; les modèles hiérarchique stationnaire permettant un inférence rapide par coupe de graphe [4]. Ces travaux sont détaillés dans le chapitre 5.

Détection et reconnaissance d'objets Nos travaux sur la reconnaissance d'objets se basent essentiellement sur les modèles structurés et semi-structurés, partiellement en lien avec les projets ANR Sattic et ANR Canada, par exemple [24, 25]. Le verrou scientifique majeur de ce domaine est l'augmentation du pouvoir de discrimination d'un modèle, tout en gardant, ou en augmentant, l'invariance vis-à-vis de transformations diverses comme les changements d'échelle, les rotations, les mouvements articulés, les changements d'éclairage etc., sachant que ces deux notions sont souvent contradictoires. L'inférence efficace reste un soucis dans ce contexte, comme pour la plupart des thématiques en vision par ordinateur. Ces travaux sont détailles dans le chapitre 4.

Reconnaissance d'actions Une partie de ces travaux est liée aux travaux sur la reconnaissance d'objets de par leurs contributions théoriques sur les modèles structurés et semi-structurés (travaux avec les doctorants Anh-Phong Ta, Mingyuan Jiu et Oya Celiktutan, ainsi que avec Prof. Bülent Sankur). Nos contributions probablement les plus notables concernent la modélisation d'activités humaines par graphes. A notre connaissance nous étions les premiers à proposer une méthode par appariement de graphes [21], une voie depuis suivie par plusieurs autres équipes. L'exemple peut être complété par nos travaux sur la solution exacte de ce problème, connu comme étant NP-complet dans le cadre général. Nous avons montré qu'une solution exacte peut être obtenue en un temps polynomial si les données sont plongées dans l'espace-temps [49]. Ensemble avec les travaux sur la reconnaissance d'objets, ces travaux sont détailles dans le chapitre 4. Certains de ces travaux sont en lien avec les projets ANR Canada et/ou ANR Sattic.

D'autres travaux sont basés sur les modèles probabilistes ou sur l'apprentissage statistique (travaux avec le doctorant Moez Baccouche et travaux avec Prof. Graham Taylor, Univ. de Guelph).

Analyse de maillages L'objectif de cette thématique est la conception de modèles de graphiques pour les modèles géométrique 3D (des maillages surfaciques) en vue de leur analyse, segmentation et filtrage. Dans un contexte de modélisation globalement cohérente, la difficulté principale est causée par la structure très irrégulière d'un maillage. Profiter pleinement de toute la richesse des données nécessite l'intégration de la géométrie, de la structure et d'éventuelles observations sur le maillage (couleurs ou textures) et souvent la prise de décisions sur ces mêmes différentes catégories d'information. Nous avons pu avancer l'état de l'art en proposant de modèles Markoviens intégrant des processus aléatoires multiples (sur les faces, sur les sommets et sur les arêtes d'un maillage) et inter-dépendant, ainsi que des

Figure 1.2 – Images tirées de la base de vidéos LIRIS-HARL tournée pour la compétition ICPR-HARL 2012.

algorithmes d'inférence. Nos méthodes ont fait preuve d'une performance plus élevée que les méthodes existantes (e.g. [2, 15]).

Ces travaux ont principalement été réalisés en collaboration avec Florent Dupont et Guillaume Lavoué du LIRIS, ainsi que avec le doctorant Vincent Vidal (recruté en tant que Maître de Conférences dans l'équipe M2Disco du LIRIS en 2012). Ils sont détaillés dans le chapitre 6. Une partie est liée au projet ANR Madras.

Evaluation L'évaluation d'algorithmes est cruciale pour l'avancement de la recherche scientifique, surtout pour les problèmes de reconnaissance d'objets ou d'actions. Très souvent il est nécessaire de concevoir des algorithmes non-triviaux pour aboutir à un mode d'évaluation satisfaisant les exigences scientifiques : (i) une interprétation simple et intuitive des mesures obtenues ; (ii) une comparaison objective entre les différents algorithmes à évaluer ; (iii) une bonne correspondance entre les mesures obtenues et la performance objective de l'algorithme à évaluer, en tenant compte du but de cet algorithme.

Nous avons proposé plusieurs solutions à ce problème pour la détection et la reconnaissance. Nous avons introduit de nouvelles mesures basées sur les mesures traditionnelles *rappel* et *précision*. La dépendance entre la mesure et les seuils est présentée sous forme de **courbes**. Ainsi, le taux de détection et la qualité de la détection peuvent être interprétés simultanément et **de façon intuitive**.

Notre algorithme pour l'évaluation d'algorithmes de détection d'objets a été utilisé pour évaluer deux compétitions scientifiques différentes, à savoir « ICDAR 2003 Robust Reading » [6] et « ImageEval 2007 » [34].

Nous considérons les compétitions scientifiques comme une opportunité formidable et un moyen puissant pour faire avancer l'état de l'art de la science dans un domaine. Pour cette raison, nous organisons actuellement la compétition scientifique ICPR HARL 2012 dans le cadre de la conférence internationale *Internationale Conference on Pattern Recognition (ICPR)*[1]. L'organisation comprend le tournage et l'étiquetage d'une base de vidéos sur les comportements humains, dont quelques images sont présentées dans la figure 1.2, la proposition d'une métrique pour l'évaluation, le classement des participants [50] et l'organisation d'une séance dédiée dans le programme de la conférence *ICPR*.

Nous avons également participé à plusieurs compétitions scientifiques, à savoir TREC 2002 pour l'indexation de la vidéo [35] et DIBCO 2009 pour la binarisation d'images. Notre algorithme de binarisation [32] a été classé $5^e/43$ lors de la compétition DIBCO [GNP11], dépassant des équipes prestigieuses comme celles de Google.

1. `http://liris.cnrs.fr/harl2012`

1.3 Les projets de recherche

Nos travaux de recherche ont été financés en partie sur des projets de recherche de différents types (voir aussi la figure 1.3 pour une illustration détaillée) :

ANR Canada — ce projet, porté par l'école des mines de Douai, avait comme but l'analyse de comportements humains dans la vidéo et la détection d'évènements anormaux. Entre 2008 et 2010 nous avons géré le partenaire LIRIS de ce projet, qui a financé, en partie, la thèse d'Anh-Phuong Ta. Ce projet a marqué le point de départ de notre intérêt pour la modélisation et la reconnaissance d'activités humaines, un sujet qui est toujours au cœur de nos préoccupations.

ANR SaTTiC — l'objectif de ce projet, porté par le laboratoire LHC de St. Etienne, était l'étude et la conception de modèles structurés et semi-structurés (graphes, chaînes, arbres etc.) pour la classification d'images et de vidéos. Ce sujet théorique vise à combiner la puissance des méthodes issues de l'apprentissage statistique avec la richesse des modèles structurés et avec la théorie de graphes pour les applications en traitement d'images et en vision par ordinateur. Les connaissances obtenues dans le cadre de ce projet nous ont également permis de résoudre certains problèmes qui se sont posés dans le cadre du projet ANR Canada. Une suite à ce projet est en cours de rédaction.

ANR Madras — la segmentation, le traitement et l'analyse de maillages surfaciques 3D étaient les principaux sujets de ce projet porté par Florent Dupont du LIRIS. Il nous a permis de combiner nos connaissances sur la modélisation globalement cohérente par modèles graphiques probabilistes avec nos connaissances sur la modélisation géométrique. Nos travaux en cours tentent d'étendre ces travaux à l'analyse de séquences vidéos issues de capteurs de profondeur (Kinect).

INTERABOT — débutant en septembre 2012, ce projet témoigne de notre ambition d'étendre nos activités sur la vision intelligente à la robotique mobile. Porté par l'entreprise « Awabot », ce projet de type « Investissements d'Avenir » a comme objectif de rendre plus intelligent et plus interactif le robot mobile « Emox » de l'entreprise, en le dotant de capacités cognitives : reconnaissance de gestes, d'objets, d'activités etc. Un aspect important sera le traitement de vidéos issues de capteurs de profondeur.

1.4 Collaborations

La figure 1.4 montre mes collaborations au sein du LIRIS, au niveau national et niveau international. Faute de place, je détaillerai seulement les dernières :

Graham Taylor est eq. maître de Conférences à l'Université de Guelph, Canada, et spécialiste en apprentissage statistique par modèles structurés. Nous avons collaboré sur la modélisation d'activités humaines par machines de Boltzmann restreintes et sur l'apprentissage par renforcement. M. Taylor a fait deux séjours à Lyon et d'autres collaborations sont prévues pour l'avenir. Publications communes : [51] [30]

Bülent Sankur est professeur à l'Université de Bogacizi à Istanbul, Turquie, et spécialiste en traitement de signal et biométrie. Ensemble nous co-encadrons Oya Celiktutan, actuellement en 2é année de thèse, sur l'appariement de graphes issus de données spatio-temporelles avec des applications sur la reconnaissance d'actions. Publications communes : [36] [49]

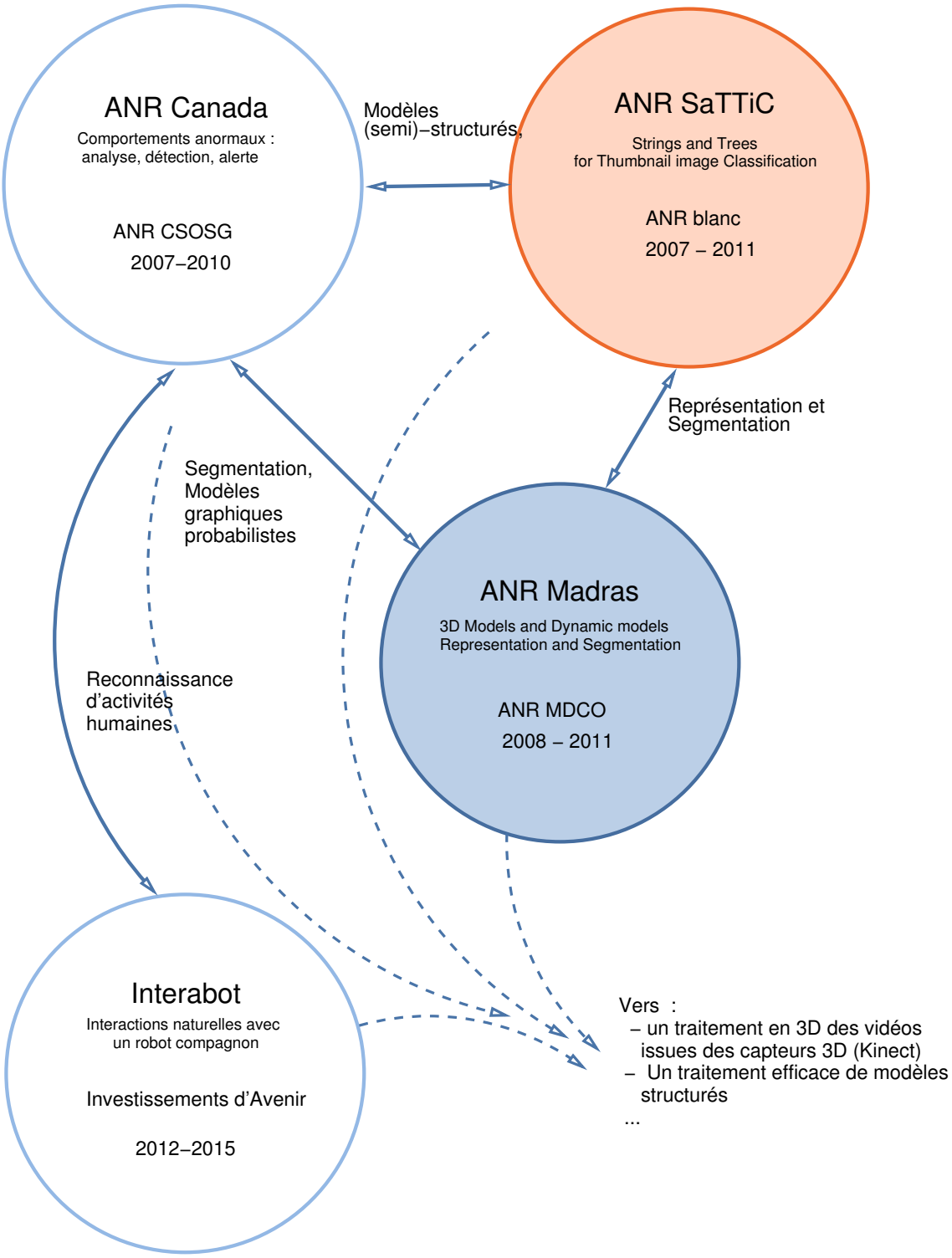


Figure 1.3 – Les projets de recherche.

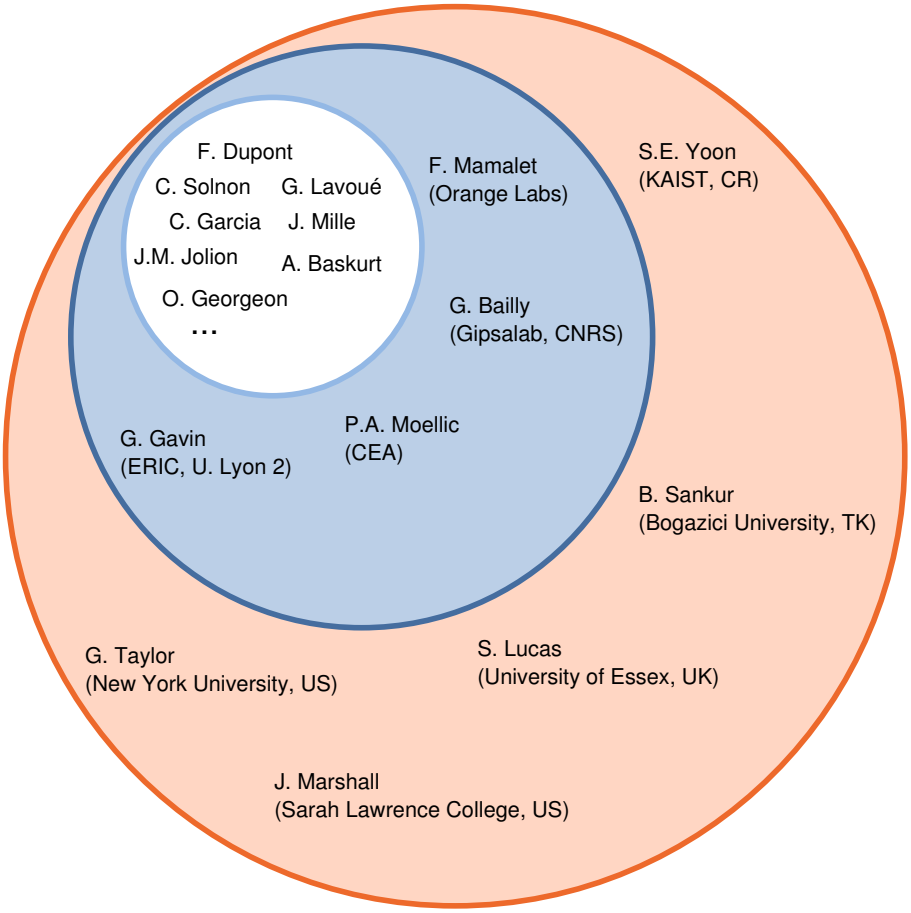


Figure 1.4 – Les collaborations (post - thèse), au sein du laboratoire, au niveau national et international.

Sung-Eui Yoon est professeur au KAIST en Corée et spécialiste sur le passage à l'échelle pour la modélisation géométrique. Nous avons collaboré sur l'analyse et le traitement d'images dites « giga-pixels », donc des images de très grandes tailles demandant des algorithmes spécifiques. Ensemble nous avons encadré Pierre-Yves Laffont, alors en stage M2R. Publication commune : [23]

Simon Lucas est professeur à l'Université d'Essex, UK, et spécialiste en reconnaissance de formes et apprentissage. En 2003, Simon Lucas a organisé la compétition scientifique "ICDAR 2003 Robust reading". Notre collaboration portait sur l'utilisation de notre métrique d'évaluation [5] pour l'évaluation des participations de cette compétition. Publication commune : [6]

James Marshall est professeur au Sarah Lawrence College, USA, et spécialiste en modélisation et apprentissage de comportements, surtout appliqué à la robotique mobile. Ensemble avec Olivier Georgeon du LIRIS, nous collaborons actuellement sur la conception d'algorithmes d'apprentissage de comportements à partir de données visuelles issues de capteurs de profondeur.

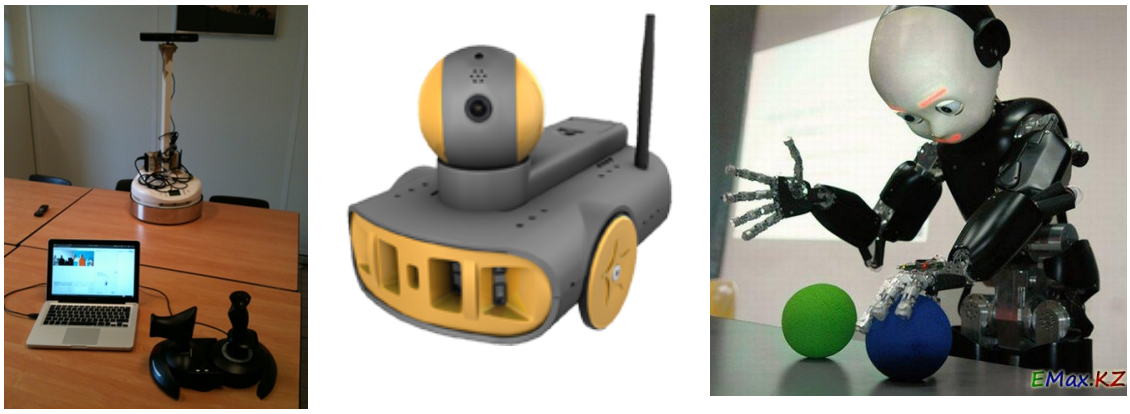


Figure 1.5 – Vers la robotique mobile : (a) la plateforme « VOIR » *Vision and Observation In Robotics* du LIRIS ; (b) le robot EMOX développé par Awabot ; (c) le robot ICUB2.

1.5 La suite : vision intelligente et robotique mobile

Depuis un an nous orientons une partie de nos recherches vers la robotique mobile. Cela est motivé à la fois par le dynamisme de ce domaine — la robotique est en plein essor — aussi bien que par la nature de nos travaux historiques, lesquels pourraient profiter pleinement d'un cadre applicatif comme la robotique mobile et d'une plate-forme associée. Nos efforts pour concentrer ces activités autour de l'application robotique mobile se manifestent par plusieurs actions actuellement en cours :

- la création de la plate-forme « VOIR » (*Vision and Observation In Robotics*) autour de trois robots mobiles équipés de caméras de profondeur [2] — voir aussi la figure 1.5a.
- le démarrage de deux thèses dans ce domaine (voir aussi la section 1.6).
- la collaboration avec Awabot, une entreprise de robotique mobile de la région Lyonnaise développant une plateforme de robotique ludique (cf. la figure 1.5b). Le projet INTERABOT temoigne de cette collaboration.
- la proposition d'un projet FP7 sur la robotique mobile, en collaboration avec plusieurs partenaires académiques et industriels. Ce projet est actuellement en cours d'expertise.

L'association de nos activités historiques autour d'une application concrète, ç.à.d. la robotique mobile, pourrait nous offrir plusieurs opportunités. Entre autres, cela nous permettra

- de poursuivre nos recherches dans un environnement plus difficile, ç.à.d. de développer et de tester nos algorithmes de reconnaissance sur un matériel réel et dans des conditions réalistes.
- de profiter d'informations supplémentaires issues des capteurs du robot (odomètres, capteurs de distance et.) afin d'améliorer les performances de nos algorithmes.
- de nous placer dans un contexte de vision active, c.à.d. de pilotage d'algorithmes de vision par un contrôle actif du capteur, c.à.d. du robot. A titre d'exemple on peut évoquer un robot changeant sa position pour mieux reconnaître un objet ou un acteur.
- de travailler sur des algorithmes de reconnaissance par intégration de plusieurs vues.
- d'étendre nos travaux sur la modélisation de comportements humain pour passer de la

2. `http://liris.cnrs.fr/voir`

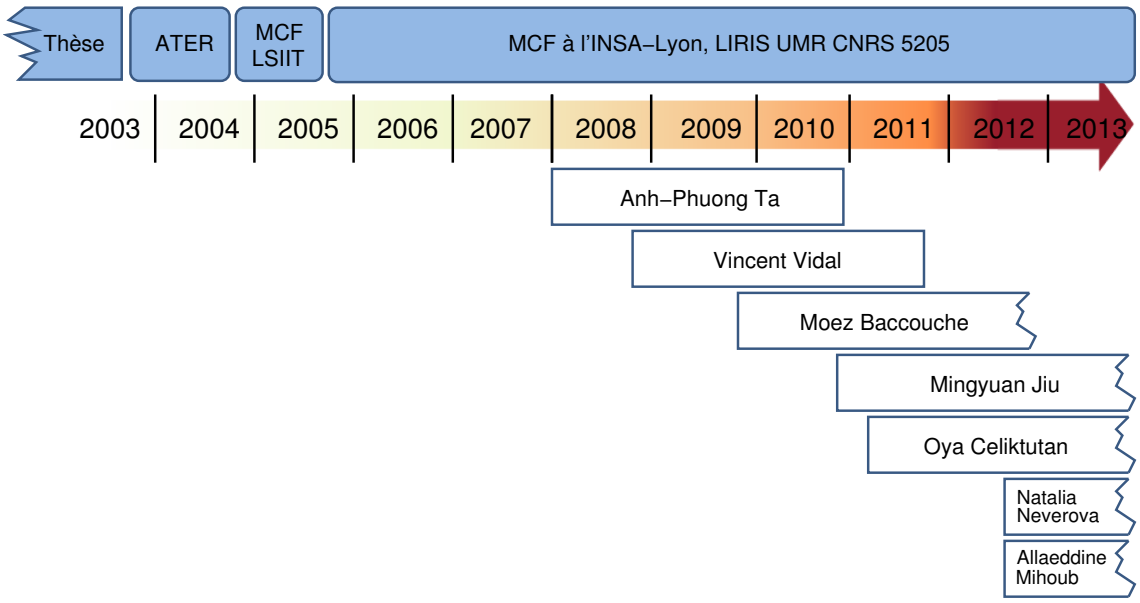


Figure 1.6 – Historique des fonctions (en haut) et encadrements doctoraux (en bas).

reconnaissance vers l'inter-action. Une thèse sur ce sujet démarrera en octobre 2012 en collaboration avec le Gipsalab, qui est actuellement en cours d'acquisition d'un robot de type ICUB2 (illustré dans la figure 1.5c).

Plus de détails sur les perspectives de nos travaux, surtout au niveau théorique et méthodologique, sont donnés dans le chapitre 7.

1.6 Activités d'encadrement de recherche

Depuis 2008, nous avons encadré 5 doctorants : 2 thèses ont été soutenues, 3 thèses sont en cours. Deux autres thèses démarreront en octobre 2012. Un de nos anciens doctorants a pris ses fonctions de Maître de Conférences à l'Université Lyon 1 à partir de septembre 2012. Le tableau 1.6 résume les thèses avec leurs financements ainsi que les équipes d'encadrement et les publications qui en ont découlés. La liste de nos publications est donnée dans la section 1.10. La figure 1.6 illustre l'historique des encadrements doctoraux.

Doctorants

Anh-Phuong Ta (thèse soutenue)

Début de thèse :	1.2.2008
Soutenance :	26.11.2010 (durée $<$ 36 mois)
Financement :	CIFRE avec Pinka Studio ; ANR Canada
Encadrants :	Christian Wolf, Guillaume Lavoué, Atilla Baskurt (Prof)
Situation actuelle :	Post-doctorant à l'INRIA Rennes
Sujet :	Mise en correspondance de graphes : application à la détection d'objets et d'activités dans la vidéo.

Cette thèse nous a permis de traiter les points communs entre la reconnaissance d'objets non-rigides et la reconnaissance d'activités humaines. Partant d'une description structurée d'un objet ou d'une action, nous avons proposé des modèles structurés et semi-structurés permettant des appariements efficaces et une puissance de classification élevée [21, 24, 19].

Vincent Vidal (thèse soutenue)

Début de thèse :	1.10.2008
Soutenance :	19.12.2011
Financement :	Allocation ministérielle fléchée sur le sujet
Encadrants :	Christian Wolf, Florent Dupont (Prof)
Situation actuelle :	Maître de Conférences à l'Univ. Lyon 1 depuis le 1.9.2012
Sujet :	Développement de modèles graphiques probabilistes pour analyser et remailler les maillages triangulaires 2-variété.

L'objectif de cette thèse était de profiter de la puissance des modèles graphiques probabilistes pour des problèmes d'analyse et de traitement de modèles géométriques 3D, c.à.d. des maillages surfaciques. Dans ce contexte, la richesse des données est à la fois une opportunité et un défi. Nos contributions permettent une modélisation globalement cohérente des différents problèmes, toute en permettant l'inférence efficace [2, 15, 17, 26].

Moez Baccouche (thèse en cours)

Début de thèse :	1.10.2009
Soutenance :	prévue pour début 2013
Financement :	CIFRE avec Orange Labs
Encadrants :	Franck Mamalet, Christian Wolf, Christophe Garcia (Prof), Atilla Baskurt (Prof)
Titre :	Classification automatique et neuronale de séquences vidéo

Partant d'un cadre applicatif autour de la reconnaissance d'actions, cette thèse, dont la soutenance est prévue pour fin 2012, vise des solutions basées entièrement sur l'apprentissage automatique. La difficulté réside dans l'apprentissage conjoint d'un modèle séquentiel et d'un modèle permettant d'extraire des caractéristiques de niveau intermédiaire [13, 14, 18, 22, 38, 50].

Mingyuan Jiu (thèse en cours)
Début de thèse : 1.10.2010
Financement : Bourse CSC (Chinese scholarship council)
Encadrants : Christian Wolf, Atilla Baskurt (Prof),
Sujet : Analyse de scènes complexes et reconnaissance d'activités
Cette thèse traite le problème de la reconnaissance d'activités complexes (activités de longues durées, interactions homme-homme, interactions homme-objet). La nature du problème rend très difficile les approches sur l'apprentissage pur, nécessitant une modélisation structurée [1, 37, 50].

Oya Celiktutan (thèse en cours)
Début de thèse : 1.1.2011
Financement : Bourse Turque
Encadrants : Bülent Sankur (Prof, Istanbul), Christian Wolf
Sujet : Reconnaissance d'actions par appariement de graphes
Toujours dans un contexte de reconnaissance d'actions dans les vidéos, cette thèse traite ce problème en modélisant les activités par des graphes. Nous nous intéressons particulièrement à l'appariement de graphes et à l'appariement par graphes [49, 50, 16]. La thèse fait partie de nos collaborations avec Bülent Sankur de l'Université de Bogacizi, Istanbul, Turkie.

Natalia Neverova (thèse démarrant)
Début de thèse : 1.10.2012
Financement : Projet INTERABOT (appel « Investissements d'Avenir »)
Encadrants : Christophe Garcia (Prof), Christian Wolf
Sujet : Interactions hommes-robots : détection et reconnaissance visuelle
Démarrant en octobre 2012, cette thèse renforcera les collaborations déjà existantes du LIRIS avec la société Awabot. La problématique est en lien avec les travaux en cours sur la reconnaissance d'activités (thèses de M. Baccouche, de M. Jiu et de O. Celiktutan), ici portant spécifiquement sur la robotique mobile et sur l'exploitation de la profondeur.

Allaeddine Mihoub (thèse démarrant)
Début de thèse : 1.10.2012
Financement : Bourse région RA (ARC 6)
Encadrants : Gerard Bailly (DR CNRS, Gipsalab), Christian Wolf
Sujet : Attention et communication homme-robot dans des tâches de co-manipulation
Ce projet est le fruit d'un collaboration démarrant entre le LIRIS et le Gipsalab sur la modélisation des interactions homme-homme et homme-robot. L'objectif est de combiner l'expertise au LIRIS sur la reconnaissance d'activités visuelles et l'expertise au Gipsalab sur les modèles d'attention et la modélisation des inter-actions verbales et co-verbales.

Étudiants du 2^e cycle en stage de recherche (M2R etc.)

Les tableau suivant recense nos encadrements de stages de recherche de niveau master 2 recherche ou équivalent. Les stages sans contexte recherche ont été omis.

Nellie Cardot, Mathieu Barralon	PFE INSA-TC	2009
Reconnaissance d'objets par appariement de graphes		
Pierre-Yves Laffont	M2R	2009
Un opérator de zoom tenant compte du contenu d'une image [23]		
Vanamali T.P	IIIT Indien	2009
Détection de comportements anormaux dans les séquences vidéo		
Quentin Bonnard	M2R	2008
Classification d'images par réseaux Bayesiens		
Graham Taylor	M2R	2004
Apprentissage de paramètres par renforcement [30] ; situation actuelle : eq. Maître de Conférences à l'Univ. de Guelph, Canada.		

1.7 Activités d'enseignement

Durant ma carrière j'ai enseigné l'informatique et la vision par ordinateur dans 2 établissements différents : l'école de physique de Strasbourg et l'INSA de Lyon :

Période	Fonction	Etablissement
2005 - présent	MCF	INSA-Lyon, Départements IF et PC
2004 - 2005	MCF	ENSPS (Ecole Nat. Sup. de Physique de Strasbourg)
2003 - 2004	ATER (temps plein)	INSA-Lyon, Département GI
2002 - 2003	Vacataire	INSA-Lyon, Département GI
2001 - 2002	Vacataire	INSA-Lyon, Département GI

L'enseignement est un travail — une vocation — qui nous demande de gérer des responsabilités diverses. Dés le début de mes enseignements, je me suis interrogé sur la pédagogie à suivre. Dans les établissements d'enseignement supérieur nous formons des futurs chercheurs, cadres, chefs de production, chefs d'entreprise etc. Nous préparons des hommes et des femmes qui doivent répondre aux exigences de la société moderne. On attend de nos jeunes diplômés qu'ils sachent s'intégrer dans une équipe, travailler de façon autonome, prendre des décisions et agir en toute responsabilité.

Comment peut-on préparer au mieux les étudiants à faire face aux exigences du monde industriel ? La simple transmission des informations ne saurait être suffisante, puisque l'acquisition des connaissances se fait par soi-même. C'est pourquoi mes enseignements ont été organisés afin de favoriser la recherche de la solution autant que la solution elle-même.

A titre d'exemple je cite (i) les mini-projets en 2^e année du premier cycle de l'INSA-Lyon, dont les cahiers des charges restent assez ouverts, favorisant un apprentissage par le plaisir et par la découverte ; (ii) les séminaires sur l'analyse de la vidéo en 5^e année au département informatique, où j'ai introduit des séances pratiques sur machine accompagnant les séances théoriques ; (iii) le module « architecture des ordinateurs » en 3^e année, où l'amélioration du contenu est réalisée en collaboration avec les étudiants.

Durant ma carrière j'ai été amené à mettre en place plusieurs enseignements :

— Lors de mon recrutement comme Maître de Conférences à l'école de physique de Strasbourg en 2004, j'ai été le seul informaticien de l'équipe pédagogique. Mon rôle a donc été naturellement de prendre en charge la nouvelle option « informatique » de l'école, introduite dans la même année, et de monter les enseignements les plus importants.
— Lors de mon année d'ATER au département GI de l'INSA de Lyon en 2003/2004, j'ai repris le cours *Vision Industrielle*. J'ai adapté cet enseignement à un support électronique et pédagogique, ce qui a été l'occasion d'une mise à jour et d'une amélioration de son contenu.
— Ensemble avec Guillaume Beslon et Marine Minier, l'enseignement de l'architecture des ordinateurs a été repris en introduisant un module important sur la conception d'un microcontrôleur basé sur la simulation.
— L'enseignement de la programmation en 2^e année du premier cycle est actuellement repris en changeant le langage de programmation. En tant que responsable de la 2^e année, je supervise ces activités.

Tous mes supports de cours sont en ligne sur mon site web, qu'il s'agisse d'enseignements toujours actualité ou d'anciens enseignements [3].

Néanmoins, un support électronique n'est pas en soi la garantie d'un enseignement de meilleure qualité. En effet, dans mes enseignements, j'ai mis l'accent sur l'utilisation de moyens modernes (toute proportion gardée, bien sûr) combinés avec des techniques pédagogiques traditionnelles. Très souvent, il est préférable de passer par un moyen pédagogique ayant déjà fait ses preuves, pour présenter des concepts plus complexes. Par exemple, exposer clairement le problème au tableau laisse à l'étudiant le temps de s'imprégner lentement des étapes de la solution. Je pratique ce mode traditionnel en préférence au premier cycle, où, de mon expérience, l'introduction des bases de la programmation profite particulièrement d'une présentation pédagogique et moins rapide sur un tableau traditionnel. Certains TD au département informatique peuvent également être adaptés à ce genre d'enseignement.

L'INSA de Lyon insiste sur une grande *ouverture à l'internationale* — 30% des étudiants en 2^e cycle sont des étrangers, 50% en 3^e cycle. Etant d'origine étrangère moi même, j'ai profité de mes connaissance du système éducatif Autrichien et de mes contactes à l'Université de Technologie de Vienne pour monter un cursus intégré (un double diplôme) en informatique entre l'INSA de Lyon et cette dernière [4]. Ce parcours, ouvert aux étudiants Français ainsi qu'aux étudiants Autrichiens, conduit à l'obtention des titres de Diplom-Ingenieurin / Diplom-Ingenieur de l'Université de Technologie de Vienne (TU) (Traduction internationale : "Master of Sciences, MSc") et d'ingénieur diplômé de l'Institut National des Sciences Appliquées de Lyon (INSA), spécialité Informatique, sous conditions d'un séjour d'au moins 4 semestres au sein de l'établissement partenaire.

L'enseignement en général, et plus spécifiquement dans le cadre d'un TP, demande une grande adaptation vis-à-vis des étudiants. Tous n'ayant pas le même niveau scolaire, des stratégies pédagogiques différentes doivent être mises en oeuvre. Gardant pour objectif de guider chaque étudiant dans sa démarche d'apprentissage, je me suis donc adapté au besoin de chacun. Une grande palette d'outils pédagogiques (techniques visuelles, comparatives etc.) et la volonté de comprendre les limites de chacun sont indispensables dans ce contexte.

La liste suivante résume mes activités d'enseignement. Les encadrements de PFE et d'autres stages

3. `http://liris.cnrs.fr/christian.wolf/teaching`
4. `http://liris.cnrs.fr/christian.wolf/doublediplome`

(rémunérés par certains départements) ont été omis de la liste :

Établiss.	Niveau	Année(s)	h(eq.TD)	Type	Matière
INSA-PC	B+2	2007-pr.	∼100h	CM	Algorithmie, programmation en pascal, projets
INSA-PC	B+2	2005-pr.	∼700h	TD/TP	Algorithmie, programmation en pascal, projets
INSA-IF	B+3	2005-pr.	∼500h	TD/TP	Architecture d'ordinateur ; conception et simulation d'un micro-contrôleur
INSA-IF	B+3	2005-2009	160h	TD/TP	Differents modules en réseaux : couches OSI, automates etc.
INSA-IF	B+5	2010-pr.	24h	SEM	Analyse de la vidéo, vision
INSA-TC	B+4	2010-pr.	8h	CM	Analyse de la vidéo, vision
ENSPS	B+4	2004-2005	16h	CM/TP	Bases de données
ENSPS	B+5	2004-2005	16h	CM/TP	Bases de données avancées
ENSPS	B+4	2004-2005	12h	TP	Acquisition d'images
ENSPS	B+4	2004-2005	32h	TP	Programmation orientée objet
INSA-GI	B+4	2002-2005	128h	CM/TP	Vision industrielle
INSA-GI	B+3	2002-2004	36h	TP	Probabilité et Statistique
INSA-GI	B+3	2001-2004	∼190h	TP	Programmation, multi-tâche, réseau
Somme			∼1900h	(non inclus : encadr. PFE, tutorats, PRP)	
Nombre moyen par année :			∼240h	(toutes rémunérations comprises)	

1.8 Responsabilités diverses

Membre de comités de sélection pour le recrutement de MCF :
- CNU 27, MCF-0026, IUT de St. Dié + LORIA, Nancy (2012)
- CNU 27, MCF-0391, Univ. Jean Monnet + LHC, St.Etienne (2012)
- CNU 27, MCF-0130, ENSEEIHT + IRIT, Toulouse (2011)
- CNU 61, MCF-0827, Télécom St. Etienne + LHC, St. Etienne (2011)

Membre de jurys de thèses :
- Thibalut LeLore, Toulon, 2011
- Emilie Samuel, St. Etienne, 2011

Responsable adjoint de l'équipe *Imagine* du LIRIS depuis octobre 2012.

Responsable de la 2^e année en Informatique au premier cycle de l'INSA, filiaires classiques et « Science en Anglais » (∼ 600 étudiants) : gestions des heures, des remplacements, des interrogations et des devoirs de synthèse, de la mise à jour des contenu (changement intégral du programme en 2012/2013) etc.

Montage et gestion d'un cursus intégré (double-diplôme) en informatique entre l'INSA de Lyon et la TU Vienne (Autriche).

Membre du vivier interne d'experts pour la constitution des comités de sélection (CNU 61). Membre externe de 4 comités de sélection pour le recrutement des Maître de Conférences (CNU 27 et 61).

Organisation de la compétition scientifique ICPR HARL 2012 dans le cadre de la conférence internationale *International Conference on Pattern Recognition (ICPR)*.

Rapporteur pour les revues internationales IEEE-Transactions on Pattern Analysis and Machine Intelligence (PAMI), IEEE-Transactions on Image Processing, IEEE-Transactions on Systems, Man and Cybernetics Serie B, Computer Vision and Image Understanding (CVIU), Pattern Reconition (PR), Pattern Recognition Letters (PRL), The Visual Computer, International Journal on Document Analysis and Recognition (IJDAR) et pour les conférences internationales Asian Conference on Computer Vision (ACCV), EUSipCo, Graphics Interface, CBDAR.

Membre du comité de programme pour les conférences internationales CBDAR 2007 et CBDAR 2009 et les conférences nationales CORESA 2009 et CORESA 2010.

Responsable de la commission bibliothèque du LIRIS ; correspondant documentaire du LIRIS auprès de l'INIST ; Membre de la commission de thèses du LIRIS ; Responsable web de l'équipe IMAGINE du LIRIS.

Gestion du partenaire LIRIS du projet ANR Canada et du partenaire LIRIS du projet INTER-ABOT (Projet d'investissement d'avenir).

1.9 Les travaux les plus cités

Source : « Google scholar ». Actualisée le 29.9.2012 [5].

Nb. de citations	Référence	Remarques
111	[32]	Méthode classée 5/43 lors de la compétition DIBCO 2009.
65	[7]	
57	[33]	
56	[6]	
41	[5]	
38	[31]	
26	[3]	
14	[48]	
11	[29]	
10	[56]	
...		

5. Voir `http://scholar.google.com/citations?user=idYS1AIAAAAJ` pour la liste actuelle

1.10 Liste de publications

Revues internationales

[1] M. Jiu, C. Wolf, C. Garcia, and A. Baskurt. Supervised learning and codebook optimization of bag of words models. *Cognitive Computation*, 4 :409–419, 2012.

[2] V. Vidal, C. Wolf, and F. Dupont. Combinatorial mesh optimization. *The Visual Computer*, 28(5) :511–525, 2012.

[3] C. Wolf. Document ink bleed-through removal with two hidden markov random fields and a single observation field. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(3) :431–447, 2010.

[4] C. Wolf and G. Gavin. Inference and parameter estimation on hierarchical belief networks for image segmentation. *Neurocomputing*, 43(4–6) :563–569, 2010.

[5] C. Wolf and J.-M. Jolion. Object count/Area Graphs for the Evaluation of Object Detection and Segmentation Algorithms. *International Journal on Document Analysis and Recognition*, 8(4) :280–296, 2006.

[6] S.M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto, H. Miyao, J. Zhu, W. Ou, C. Wolf, J.-M. Jolion, L. Todoran, M. Worring, and X. Lin. ICDAR 2003 Robust Reading Competitions : Entries, Results and Future Directions. *International Journal on Document Analysis and Recognition - Special Issue on Camera-based Text and Document Recognition*, 7(2-3) :105–122, 2005.

[7] C. Wolf and J.-M. Jolion. Extraction and Recognition of Artificial Text in Multimedia Documents. *Pattern Analysis and Applications*, 6(4) :309–326, 2003.

Brevets

[8] C. Wolf, J.M. Jolion, and C. Laurent. Détermination de caractéristiques textuelles de pixels. Brevet France Télécom, No. FR 03 11918, October 2003.

[9] C. Wolf, J.M. Jolion, and F. Chassaing. Procédé de détection de zones de texte dans une image vidéo. Brevet France Télécom No. FR 01 06776, June 2001.

Conférences invitées

[10] C. Wolf and A. Baskurt. Action recognition in videos. Invited talk at International Conference on Image Processing Theory, Tools and Applications, Istanbul, 2012.

[11] C. Wolf. Maintien des personnes âgées à domicile - enjeux scientifiques et technologiques liés à la vision par ordinateur. Conférence invitée à l'école d'été "Intelligence ambiante", Lille, 2011.

Conférences internationales

[12] M. Jiu, C. Wolf, and A. Baskurt. Integrating spatial layout of object parts into classification without pairwise terms : application to fast body parts estimation from depth images. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2013.

[13] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Spatio-temporal convolutional sparse auto-encoder for sequence classification. In *British Machine Vision Conference (BMVC)*, 2012.

[14] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sparse shift-invariant representation of local 2d patterns and sequence learning for human action recognition. In IEEE, editor, *International Conference on Pattern Recognition (ICPR)*, 2012.

[15] V. Vidal, C. Wolf, and F. Dupont. Mesh segmentation and global 3d model extraction. In *Poster at Symposium on Geometry Processing (SGP)*, 2012.

[16] O. Celiktutan, C. Wolf, and B. Sankur. Real-time exact graph matching with application in human action recognition. In *International Workshop on Human Behavior Understanding*, 2012.

[17] V. Vidal, C. Wolf, and F. Dupont. Robust feature line extraction on cad triangular meshes,. In *Proceedings of the International Conference on Computer Graphics Theory and Applications*, 2011.

[18] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding : Inducing Behavioral Change*, 2011.

[19] A.P. Ta, C. Wolf, G. Lavoué, A. Baskurt, and J-M. Jolion. Pairwise features for human action recognition. In *Proceedings of the International Conference on Pattern Recognition*, 2010.

[20] C. Wolf and J.M. Jolion. Integrating a discrete motion model into GMM based background subtraction. In *Proceedings of the International Conference on Pattern Recognition*, 2010.

[21] A.-P. Ta, C. Wolf, G. Lavoue, and A. Baskurt. Recognizing and localizing individual activities through graph matching. In *International Conference on Advanced Video and Signal-Based Surveillance (Best paper for track "recognition")*, 2010.

[22] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Action classification in soccer videos with long short-term memory recurrent neural networks. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 154–159, 2010.

[23] P.Y. Laffont, J.Y. Jun, C. Wolf, Y.W. Tai, K. Idrissi, G. Drettakis, and S.-E. Yoon. Interactive content-aware zooming. In *Graphics Interface*, 2010.

[24] A.-P. Ta, C. Wolf, G. Lavoué, and A. Baskurt. 3D object detection and viewpoint selection in sketch images using local patch-based zernike moments. In *International workshop on content-based multimedia indexing*, pages 189–194, 2009.

[25] M. Mouret, C. Solnon, and C. Wolf. Classification of images based on hidden markov models. In *International workshop on content-based multimedia indexing*, 2009.

[26] V. Vidal, C. Wolf, G. Lavoué, and F. Dupont. Global triangular mesh regularization using conditional markov random fields. In *Poster at Symposium on Geometry Processing (acceptance rate 35%)*, 2009.

[27] C. Wolf. Families of markov models for document image segmentation. In *Proceedings of the IEEE Machine Learning for Signal Processing Workshop*, 2009.

[28] C. Wolf. Improving recto document side restoration with an estimation of the verso side from a single scanned page. In *Proceedings of the International Conference on Pattern Recognition*, 2008.

[29] G. Lavoué and C. Wolf. Markov random fields for improving 3d mesh analysis and segmentation. In *Proceedings of the EUROGRAPHICS 2008 Workshop on 3D Object Retrieval*, 2008.

[30] G.W. Taylor and C. Wolf. Reinforcement Learning for Parameter Control of Text Detection in Images and Video Sequences. In *Proceedings of the International Conference on Information & Communication Technologies (IEEE)*, 2004.

[31] 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.

[32] C. Wolf, J.-M. Jolion, and F. Chassaing. Text Localization, Enhancement and Binarization in Multimedia Documents. In *Proceedings of the International Conference on Pattern Recognition*, volume 2, pages 1037–1040, 2002.

[33] C. Wolf, J.M. Jolion, W. Kropatsch, and H. Bischof. Content Based Image Retrieval using Interest Points and Texture Features. In *Proceedings of the International Conference on Pattern Recognition*, volume 4, pages 234–237, 2000.

Conférences internationales sans comité de lecture

[34] C. Wolf and J.-M. Jolion. Quality, quantity and generality in the evaluation of object detection algorithms. In *Proceedings of the Image Eval Conference*, 2007.

[35] C. Wolf, D. Doermann, and M. Rautiainen. Video indexing and retrieval at UMD. In National Institute for Standards and Technology, editors, *Proceedings of the text retrieval conference - TREC*, 2002.

Conférences nationales

[36] O. Celiktutan, C. Wolf, and B. Sankur. Appariement de points spatio-temporels par hypergraphes et optimisation discrète exacte. In *COmpression et REprésentation des Signaux Audiovisuels (CORESA)*, 2012.

[37] M. Jiu, C. Wolf, C. Garcia, and A. Baskurt. Supervised learning and codebook optimization with neural network. In *COmpression et REprésentation des Signaux Audiovisuels (CORESA)*, 2012 (to appear).

[38] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Une approche neuronale pour la classification d'actions de sport par la prise en compte du contenu visuel et du mouvement dominant. In *COmpression et REprésentation des Signaux Audiovisuels (CORESA)*, 2010.

[39] C. Wolf. Séparation recto/verso d'un document par modélisation markovienne à double couche. In *Compression et Réprésentation des Signaux Audiovisuels (CORESA)*, 2009.

[40] C. Wolf and G. Gavin. Inference and parameter estimation on belief networks for image segmentation. In *Journées francophones des réseaux bayésiens*, 2008.

[41] 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 *14éme congrés francophone de reconnaissance des formes et intelligence artificielle*, 2004.

[42] C. Wolf and J.M. Jolion. Détection de textes de scénes dans des images issues d'un flux vidéo. In *Journées d'études et d'Echanges "Compression et Réprésentation des Signaux Audiovisuels"*, pages 63–66, 2003.

[43] C. Wolf and J.M. Jolion. Extraction de texte dans des vidéos : le cas de la binarisation. In *13ème congrès francophone de reconnaissance des formes et intelligence artificielle*, volume 1, pages 145–152, January 2002.

[44] C. Wolf and J.-M. Jolion. Détection et extraction de texte de la vidéo. In *Colloque International Francophone sur l'Ecrit et le Document*, pages 215–224, 10 October 2002.

[45] C. Wolf and J.M. Jolion. Vidéo ocr - détection et extraction du texte. In *Journées d'études et d'Echanges "Compression et Représentation des Signaux Audiovisuels"*, pages 251–258, 2001.

[46] C. Wolf and J.-M. Jolion. Détection et Extraction du texte de la vidéo. In *ORASIS 2001, Congrés francophone de vision, Cahors, France*, pages 415–424, 5-8 June 2001.

[47] C. Wolf, J.M. Jolion, and H. Bischof. Histograms for texture based image al. In R. Sablatnig and C. Menard, editors, *Proceedings of the OEAGM 2000*, pages 169–176. Oldenbourg, 25 May 2000.

Thèse

[48] C. Wolf. *Text Detection in Images taken from Videos Sequences for Semantic Indexing*. PhD thesis, INSA de Lyon, 2003.

Rapports de recherche

[49] O. Celiktutan, C. Wolf, and B. Sankur. Fast exact matching and correspondence with hypergraphs on spatio-temporal data. Technical Report LIRIS RR-2012-002, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2012.

[50] C. Wolf, J. Mille, E. Lombardi, O. Celiktutan, M. Jiu, M. Baccouche, E. Dellandréa, C.-E. Bichot, C. Garcia, and B. Sankur. The liris human activities dataset and the icpr 2012 human activities recognition and localization competition. Technical Report LIRIS RR-2012-004, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2012.

[51] C. Wolf and G. Taylor. Learning individual human activities from short binary shape sequences. Technical Report LIRIS RR-2011-018, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2011.

[52] V. Vidal, C. Wolf, F. Dupont, and G. Lavoué. An iterative approach for global triangular mesh regularization. Technical Report LIRIS RR-2009-032, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2009.

[53] C. Wolf and G. Gavin. Inference and parameter estimation on hierarchical belief networks for image segmentation. Technical Report RR-LIRIS-2008-21, Laboratoire d'informatique en images et systèmes d'information, 2008.

[54] C. Wolf. An iterative graph cut optimization algorithm for a double MRF prior. Technical Report LIRIS RR-2008-017, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2008.

[55] C. Wolf. Document ink bleed-through removal with two hidden Markov random fields and a single observation field. Technical Report LIRIS RR-2006-019, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2006.

[56] C. Wolf and J.-M. Jolion. Model based text detection in images and videos : a learning approach. Technical Report LIRIS RR-2004-011, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2004.

Chapitre 2

Introduction

Le tout est plus que la somme des parties — ce maxime attribuée à Aristote peut s'appliquer à quasiment tous les aspects de la vie. Comme un mot est plus que l'ensemble de ses lettres, une équipe bien soudée peut souvent réaliser plus que ce peuvent réaliser les individus séparément. Nous devons à Blaise Pascal une citation similaire :

> *Donc, toutes choses étant causées et causantes, aidées et aidantes, médiates et immédiates, et toutes s'entretenant par un lien naturel et insensible qui lie les plus éloignées et les plus différentes, je tiens impossible de connaître les parties sans connaître le tout, non plus que de connaître le tout sans connaître particulièrement les parties.*

La science s'est particulièrement intéressée à ce sujet dans le domaine de l'intelligence artificielle, où la notion de l'émergence d'une intelligence collective à partir d'une collection d'individus plus simple s'est établie. L'exemple le plus connu concerne peut-être les algorithmes d'optimisation à base de colonies de fourmis, ou en général le concept de *swarm intelligence*. Le principe est ici similaire aux exemples précédents : un résultat supérieur est obtenu en combinant, de manière intelligente et adaptée, plusieurs processus simples. La complexité du modèle ou du processus réside à la fois dans l'ensemble de ses parties, et dans leur manière d'interagir. L'émergence de complexités élevées peut alors être observées à partir d'interactions très basiques d'agents peu complexes. Il est important de noter que les interactions mêmes entre les acteurs sont souvent très simples. Autrement dit, un comportement complexe peut être obtenu à partir d'acteurs de comportements simples et d'interactions simples.

L'exemple mentionné ci-dessus fait intervenir des interactions entre agents ou, en général, entre plusieurs comportements. L'entité décomposée en plusieurs parties est donc un algorithme, un comportement. Les travaux décrits dans ce mémoire décrivent des situations où l'entité décomposée en plusieurs parties est une solution à un problème. Ils se basent sur l'hypothèse selon laquelle, dans certains cas, la solution elle même peut être modélisée de manière "collective", typiquement en combinant, de manière intelligente et adaptée, des parties plutôt simples. Comme pour les exemples ci-dessus, un bénéfice peut parfois être obtenu par cette modélisation globale. Souvent cet avantage s'explique par une cohérence globale de la solution modélisée par l'ensemble des interactions.

La vision, qu'il s'agisse du système visuel humain ou de la vision par ordinateur, profite particulièrement bien d'une telle modélisation globalement cohérente. A titre d'exemples nous pouvons citer la reconnaissance d'objets dans les images, l'application phare du domaine. D'une part, la

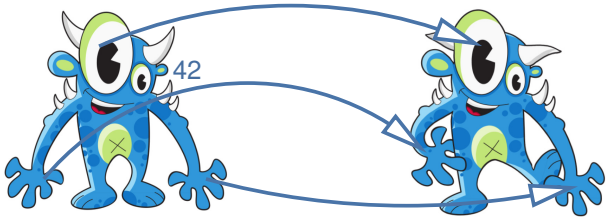


Figure 2.1 – Les objets articulés demandent la gestion d'interactions complexes entre les parties. Dessins artistiques (les deux personnages) reproduits de Soloman et al. [SBCBG11]

nature composée et souvent hiérarchique des entités nous entourant ("objets" vivants et artificiels) demande une modélisation par parties, appelée modélisation par caractéristiques locales dans ce contexte, ce qui est particulièrement intéressant si l'application requiert la gestion d'occultations d'une partie de l'objet. D'autre part, une description fidèle de l'objet, c.à.d. une description satisfaisant des propriétés d'invariance et de pouvoir de discrimination, va la plupart de temps capturer des propriétés globales de l'objet, souvent liées à la forme. Le mot "global" dans ce contexte signifie, lié à l'ensemble de l'objet ou à une grande partie.

Le respect de ces deux contraintes se traduit naturellement par une description par parties associée d'une description des interactions entre les différentes parties. Dans certains cas simples, les interactions peuvent se résumer à une équation paramétrique peu complexe ou parfois à un simple "stockage" des parties, complété par des informations comme leurs positions (relatives) dans l'image. Dans d'autres cas, les interactions peuvent s'avérer arbitrairement complexes. Si nous gardons l'exemple de la reconnaissance d'objets dans les images, la nature articulée des objets vivants (humains et animaux) rend plus complexe leur description géométrique ainsi que les invariances associées, surtout dans le cas d'une description par parties. La figure 2.1 illustre ce concept à l'aide du même personnage dessiné en deux postures différentes. Une description fidèle de l'objet doit clairement tenir compte (i) d'une grande similarité locale entre les parties respectives des deux dessins ; (ii) d'une certaine souplesse quant à la forme globale ; (iii) d'une caractérisation de géométrie non-rigide et de la topologie de l'objet.

Le concept de la description collective peut être étendu pour tenir compte d'entités non liées à des parties "physique" d'un objet. Dans le cas de la reconnaissance d'objets, une solution au problème pourrait être modélisée par une collection de variables décrivant des informations différentes, comme des parties, leurs positions relatives ; un point de vue ; d'autres informations supplémentaires comme le type et le contexte de la scène, la présence d'autres objets etc. Dans d'autres applications la description collective pourrait consister en un ensemble de variables associées au processus physique modélisé, comme c'est souvent le cas dans les modèles probabilistes de types Bayesiens (c.f. le chapitre 3.2).

La suite de ce mémoire porte principalement sur la modélisation globalement cohérente d'interactions complexes entre plusieurs variables dans le domaine de l'analyse d'images, de maillages et de vidéos, en tenant compte de critères géométriques et colorimétriques. Les algorithmes que nous avons proposé pour différentes applications servent le plus souvent

— à l'estimation d'un ensemble d'étiquettes, donc à la classification, à la segmentation et à la restauration ;
— à la mise en correspondance (matching) et à l'indexation, à l'estimation d'une transformation géométrique ;

— au *fitting*, c.à.d. l'ajustement d'un modèle paramétrique ;
— à l'estimation de mesures diverses (géométriques, mouvement etc.)

Dans ce contexte, nous avons souvent recours à une modélisation à l'aide de modèles structurés, par exemple des modèles graphiques, probabilistes ou non-probabilistes, décrivant les relations entre les différentes variables observées ou cachées du système. Fréquemment les interactions entre les variables sont modélisées par une fonction globale à minimiser ou à échantillonner, appelée *fonction d'énergie*. Les principaux verrous scientifiques sont, en général et selon l'application :

— la modélisation des interactions et des dépendances du problème en question sous forme de graphe(s), en intégrant les contraintes du domaine – souvent des phénomènes (et invariances) géométriques complexes,
— la conception d'une fonction d'énergie globale associée au(x) graphe(s),
— la conception d'un algorithme d'optimisation capable de trouver une solution globale exacte ou approchée du problème, souvent sous des contraintes fortes de faible complexité de calcul,
— la conception d'algorithmes (supervisés ou non-supervisés) d'apprentissage de paramètres.

Pour mieux motiver l'apport d'une modélisation globale et de la recherche de solutions globalement cohérentes, nous illustrons leurs avantages dans deux cas simples et concrets : la section 2.1 présentera un cas de recherche d'objets articulés dans les images et dans la section 2.2 nous parlerons de la segmentation d'images. Nous tenons à insister sur le fait que la présentation de ces exemples a été optimisée pour les lecteurs non familiers avec la modélisation globale. Nous renvoyons au chapitre 3 les lecteurs du domaine vision par ordinateur et traitement d'images, et directement aux contributions principales de ce mémoire, décrites à partir du chapitre 4, les lecteurs familiers avec les modèles Markoviens et avec la minimisation de fonctions d'énergie.

Les deux exemples suivants ont été conçus pour illustrer les avantages d'une modélisation globalement cohérente. Le choix de ces exemples a été guidé par la simplicité, au point que le calcul numérique de la meilleure solution peut être effectué sans aide d'un ordinateur ou d'une calculatrice.

2.1 Premier exemple : reconnaissance d'objets

Le problème est illustré dans la figure 2.2a : dans une image de scène, présentée à droite, il est demandé de trouver toutes les occurrences, éventuellement déformées, d'un objet dont une image modèle est donnée à gauche. Afin de pouvoir gérer des occultations partielles, la méthode utilise une décomposition en « patches », c.à.d. en fenêtres rectangulaires recouvrant les deux images de manières dense. Un descripteur d'apparence est calculé sur chaque fenêtre. Pour faciliter le calcul manuel, les descripteurs de cet exemple se résument à de simples scalaires. Cette représentation est illustrée dans la figure 2.2b.

Pour détecter et reconnaitre, une famille de méthodes classiques recourt à un appariement entre les patches du modèle et les patches de la scène. Plus précisément, chaque patche du modèle est affecté à un patch du modèle, alors que les derniers ne sont pas forcement tous cible d'une affectation. Le critère de détection et de reconnaissance est souvent directement calculé à partir de ces affectations.

Dans la suite nous discuterons plusieurs méthodes possible à ce problème d'affectation et de détection/reconnaissance. Une solution donnée du problème est caractérisée par un ensemble de valeurs pour un ensemble de variables x_i, où $x_i = j$ sera interprété comme « patche i du modèle

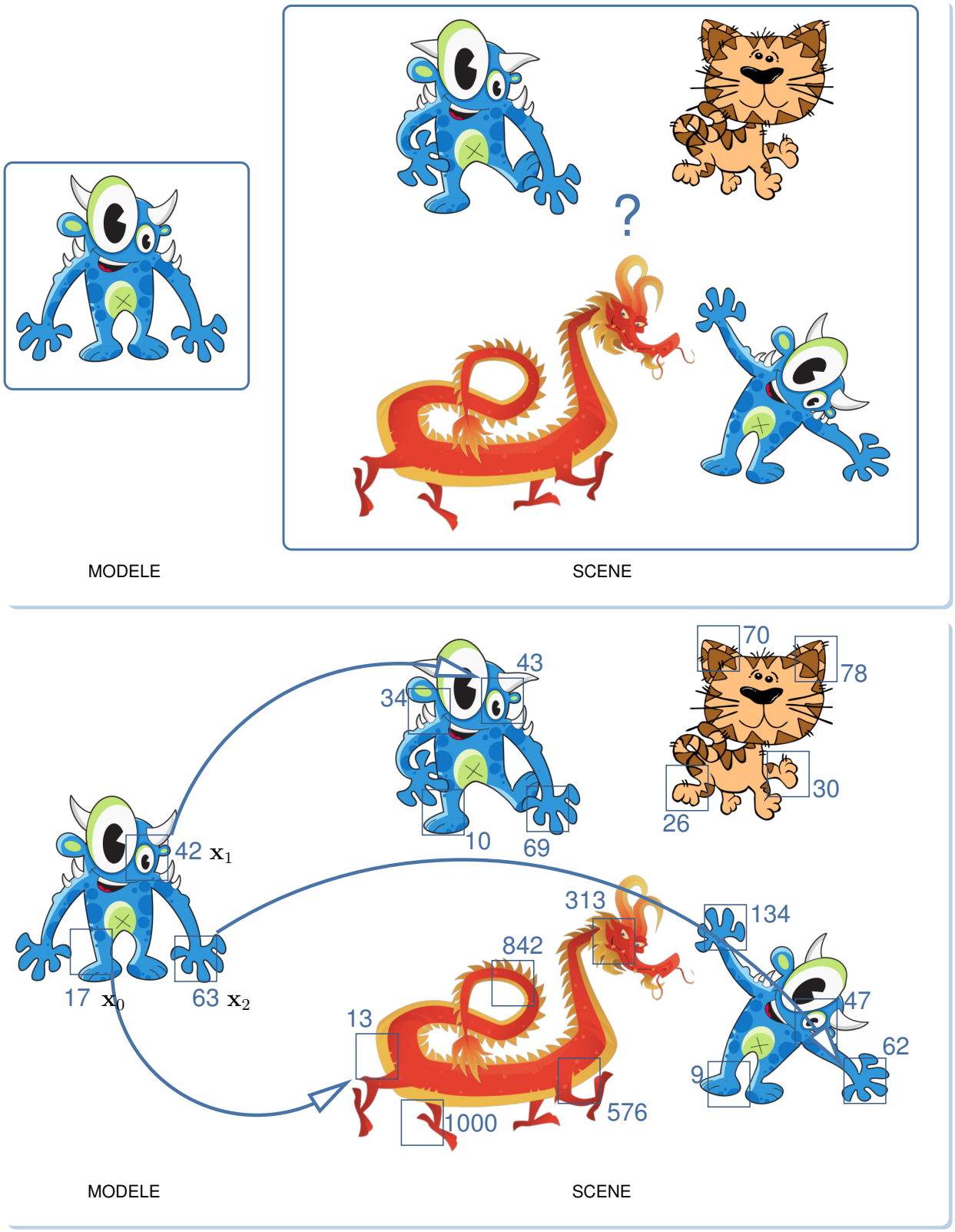


Figure 2.2 – En haut : détection et reconnaissance d'un modèle dans une scène contenant plusieurs objets. En bas : modélisation par caractéristiques locales extraites sur des patches et affectation par distances minimales entre descripteurs. Dessins artistiques (les objets) reproduits de Soloman et al. [SBCBG11]

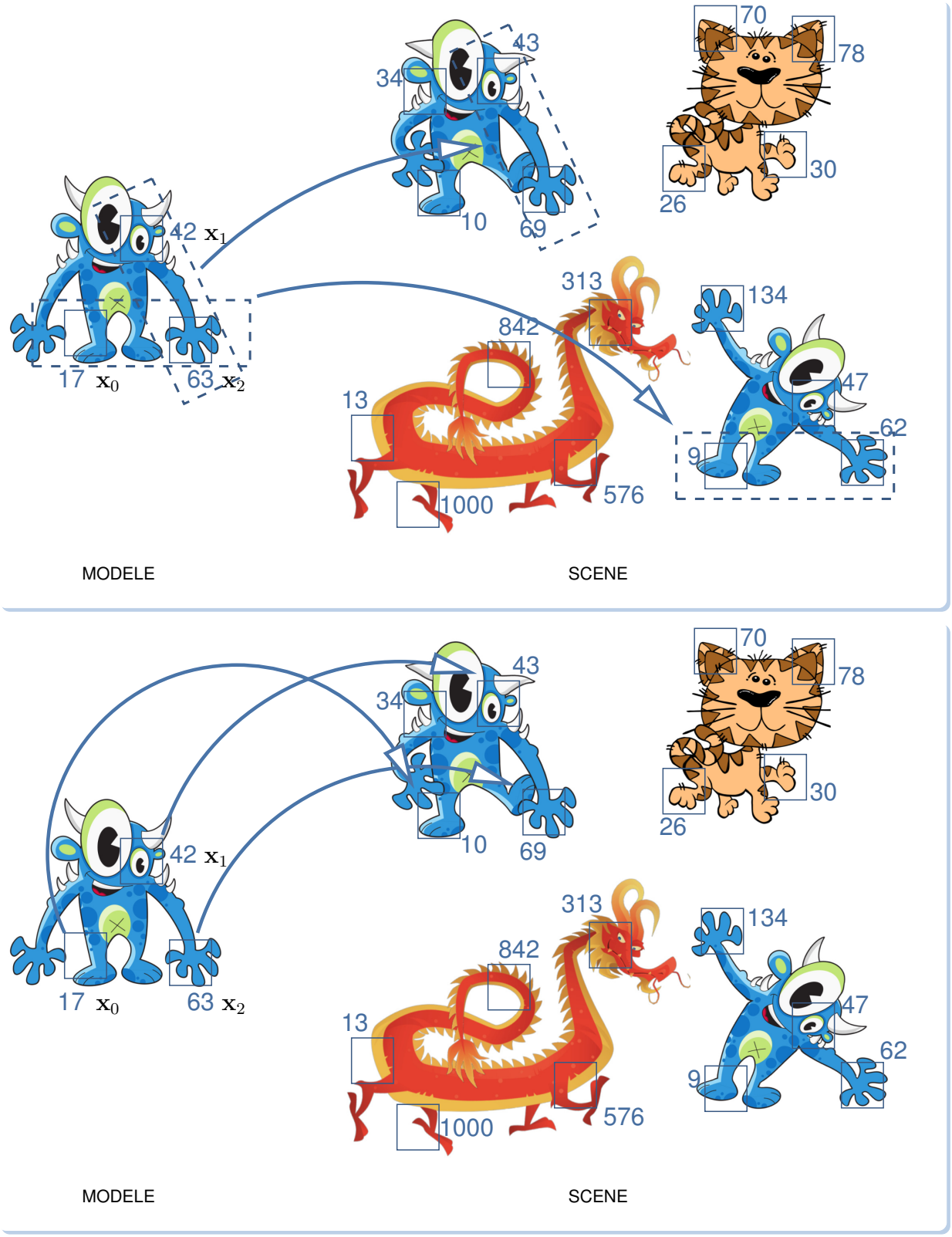


Figure 2.3 – En haut : solution 2 : affectation par paires. En bas : solution 3 : recherche de solution globale. Dessins artistiques (les objets) reproduits de Soloman et al. [SBCBG11]

est affecté au patche j de la scène ». Pour rendre la lecture plus simple, dans la suite de cette section nous nous sommes passés d'une numérotation des patches de scène. Une affectation est indiquée de manière plus simple à l'aide du descripteur de scène, c.à.d. $x_0 = 13$ indiquera une affectation du patch numéro 0 du modèle au patch de scène ayant un descripteur égale à 13 (voir la flèche partant de la jambe du personnage dans la figure 2.2). Pour enlever toute ambiguïté possible, les valeurs des descripteurs ont été choisies en évitant les doublons.

Solution 1 : affectation par descripteur — La solution la plus évidente consiste à affecter à chaque patche du modèle le patche de la scène ayant le descripteur le plus proche, c.à.d. de minimiser la distance entre les descripteurs des patches affectés :

$$\begin{aligned} x_0 &\leftarrow 13 \\ x_1 &\leftarrow 43 \\ x_2 &\leftarrow 62 \end{aligned} \tag{2.1}$$

La faiblesse de la solution est évidente : le descripteur est trop faible pour faire la différence entre les différentes parties de l'objet. Plus important, en présence de plusieurs objets du même type dans la scène, les affectations des patches modèles peuvent se distribuer sur des instances différentes dans la scène, ce qui empêche une détection et reconnaissance correcte. Il sera donc nécessaire d'ajouter des contraintes supplémentaires, typiquement en utilisant des informations sur la position des patches, ou des informations de proximité.

Solution 2 : affectation par paires — Un voisinage est créé pour le modèle et pour la scène, c.à.d. pour chaque patche nous identifions une liste de voisins directs. L'affectation est effectuée par paires de patches voisins : à chaque paire de patches voisins du modèle est affectée une paire de patches voisins de la scène telle que les deux paires se ressemblent en termes de descripteurs (en minimisant la somme des différences des descripteurs affectés sur les deux patches de la paire).

$$\begin{aligned} (x_0, x_1) &\leftarrow (10, 43) \\ (x_1, x_2) &\leftarrow (47, 62) \end{aligned} \tag{2.2}$$

Nous pouvons constater que la solution s'est améliorée : aucun patche n'a été affecté au dragon. Par contre, la solution n'est toujours pas cohérente, puisque des parties différentes du bonhomme modèle ont été affecté à des objets différents de la scène. De plus, la solution du patche x_1 (faisant partie des deux pairs (x_0, x_1) et (x_1, x_2) n'est pas cohérente, puisque il a été affecté à deux patches scènes différents.

Solution 3 : recherche de cohérence global — Les patches sont affectés d'une manière optimale sur la globalité de l'objet : sur tout l'ensemble du modèle, nous imposons la contrainte d'affecter, à chaque paire possible de deux patches voisins, deux patches de la scène également voisins. La somme, sur toutes les paires, des distances des descripteurs affectés est minimisée. Notons que le minimum global est atteint pour deux solutions différents ayant la même valeur de 14, soit

$$14 = |42 - 43| + |63 - 69| + |10 - 17| \tag{2.3}$$

obtenue par

$$x_0 \leftarrow 10$$
$$x_1 \leftarrow 43$$
$$x_2 \leftarrow 69$$

et par (2.4)

$$x_0 \leftarrow 9$$
$$x_1 \leftarrow 47$$
$$x_2 \leftarrow 62$$

La première solution correspond au bonhomme en haut à gauche, tandis que la deuxième solution correspond au bonhomme en bas à droite de la scène.

Nous pouvons constater que la méthode nr. 3 a obtenu le meilleur résultat grâce à sa stratégie de recherche favorisant des solutions globalement cohérentes. La solution obtenue correspond à un ensemble de décisions, à prendre sur des parties individuelles, tout en intégrant des contraintes de cohérences entre les différentes parties. Cela conduit nécessairement à une dépendance entre les parties et donc à une prise de décision de nature globale.

2.2 Deuxième exemple : segmentation d'images

Le deuxième exemple traite le cas d'une image issue d'une source binaire fortement bruitée, c.f. la figure 2.4. La séparation entre texte sombre et fond clair est un problème de segmentation qui est souvent réalisé à l'aide d'un algorithme de classification de pixels. C'est une tâche facile pour un humain qui s'avère également assez maîtrisable de manière automatique. Il existe toutefois quelques pièges qu'il convient d'éviter. L'agrandissement d'une zone de la figure montre la forte nature du bruit. En effet, les niveaux de gris de certains pixels de la zone de texte correspondent plutôt aux caractéristiques du fond et vice-versa.

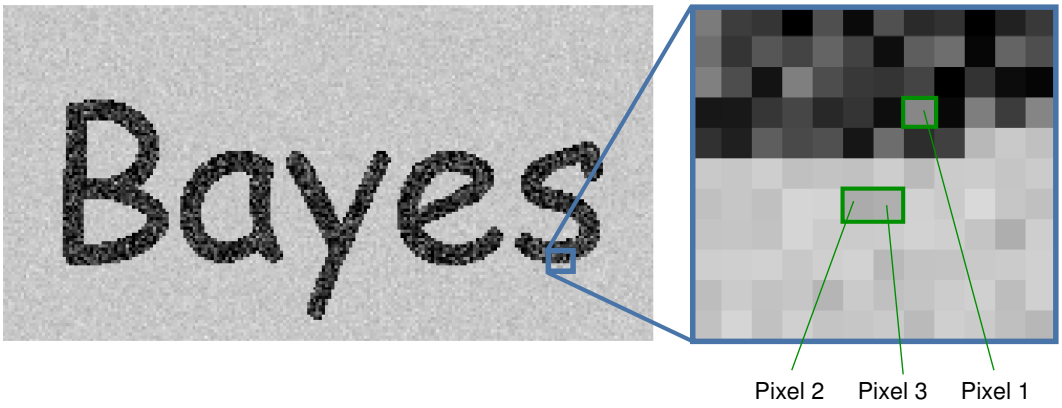


Figure 2.4 – Une image d'une source binaire ayant subi un fort bruit.

Considérons tout d'abord le cas du pixel numéro 1 de la figure 2.4 dont le niveau de gris est de $y = 135$. Pour le classifier entre texte (classe C_0) ou fond (classe C_1), nous supposons d'abord comme seule information les moyennes de niveaux de gris des deux classes, à savoir $\mu_0 = 50$ pour

la classe C_0 ainsi que $\mu_1 = 200$ pour la classe C_1. Cette situation est illustrée dans la figure 2.5a, où le niveau de gris de l'échantillon (pixel 1) est montré par une ligne pointillée.

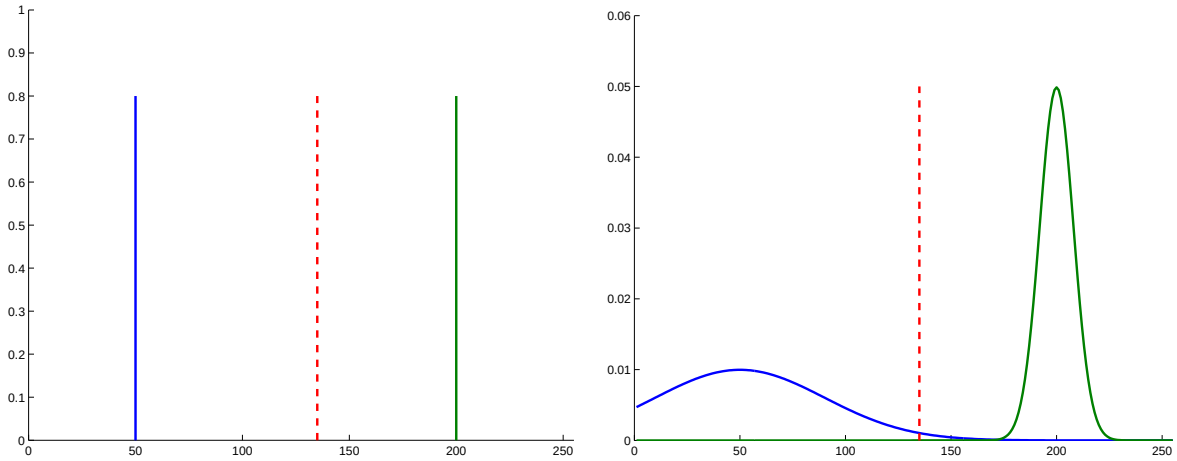


Figure 2.5 – Classification d'un échantillon : (a) par seuillage simple ; (b) en maximisant la vraisemblance.

En absence de toute information supplémentaire, la meilleure stratégie est sans doute un seuillage avec un seuil à distance égale entre les deux moyennes, à savoir $T = 125$ dans notre cas. Cela nous emmènerait à classifier l'échantillon comme appartenant à la classe C_1, $y = 135$ étant plus proche de $\mu_1 = 200$ que de $\mu_0 = 50$. Cette décision serait clairement fausse.

Le résultat peut facilement être amélioré en ajoutant d'avantage d'informations sur l'image, par exemple les variations de niveaux de gris des deux classes. Celles-ci peuvent être représentées par les écart-types respectifs des classes C_0 et C_1, à savoir $\sigma_0 = 40$ et $\sigma_1 = 8$. La situation peut maintenant être formulée de manière probabiliste en introduisant une variable aléatoire pour l'étiquette (la classe) d'un échantillon X prenant des valeurs dans l'ensemble $\{0, 1\}$, ainsi qu'une variable Y pour le niveau de gris d'un échantillon et prenant donc des valeurs dans l'ensemble $[0, 255]$:

$$p(Y = y|X = x) = \begin{cases} \mathcal{N}(y; \mu_0, \sigma_0) & \text{si } x = 0 \\ \mathcal{N}(y; \mu_1, \sigma_1) & \text{si } x = 1 \end{cases} \tag{2.5}$$

où $\mathcal{N}(y; \mu, \sigma)$ désigne la densité de probabilité d'une distribution normale de moyenne μ et de variance σ^2 évaluée pour la valeur y. La variable Y étant connue et la variable X étant cherchée, la stratégie classique dans ce genre de situation consiste à maximiser la probabilité $p(Y = y|X = x)$, appelée *probabilité conditionnelle de la classe* où plus simple, *vraisemblance*. La stratégie est connue sous le nom *maximum de vraisemblance* (ML - *maximum likelihood*) :

$$\hat{x} = \arg \max_{x \in \{0,1\}} p(Y = y|X = x) \tag{2.6}$$

Cette situation est illustrée dans la figure 2.5b, toujours sur le même échantillon de niveau de gris $y = 135$. Nous pouvons constater que les informations supplémentaires changent la décision en faveur de la classe C_0.

Notons que les symboles majuscules font ici référence à des variables aléatoires, tandis que les symboles minuscules font référence à leurs réalisations. Dans la suite de ce mémoire, et quand le

contexte le permet, nous profiterons d'une convention habituelle en dénotant les probabilités par leurs réalisations uniquement, ainsi la probabilité

$$p(X = x) \text{ sera souvent notée comme } p(x).$$

Améliorer d'avantage le résultat de classification demandera une modélisation encore plus réaliste de la situation, par exemple en améliorant la loi de vraisemblance. Pour cela une solution serait de remplacer la loi normale par une loi plus adéquate, représentant mieux la distribution des niveaux de gris des deux classes. Une autre possibilité consiste à injecter une connaissance *a priori* dans le modèle, c.à.d. une connaissance sur l'étiquette de classe x, sans tenir compte de la mesure y. Dans notre exemple d'application, nous pourrions supposer que les pixels de classe C_1 sont beaucoup plus nombreux que les pixels de la classe C_0. Cela peut être formalisé de la manière suivante :

$$\begin{aligned} p(X = 0) &= 0.3 \\ p(X = 1) &= 0.7 \end{aligned} \tag{2.7}$$

Pour intégrer cette connaissance, la décision de classification ne sera plus uniquement prise en maximisant la probabilité conditionnelle de classe. Une façon naturelle est de maximiser la probabilité *a posteriori* $p(x|y)$, c.à.d. la probabilité du résultat sachant les données observer. Cette probabilité peut être calculé en appliquant la règle de Bayes :

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)} \tag{2.8}$$

La probabilité des données $p(y)$ peut être obtenue en intégrant sur les valeurs de l'étiquettes :

$$p(y) = \sum_{x'} p(x')p(y|x')) \tag{2.9}$$

Afin de pouvoir prendre une décision sur la valeur estimée de l'étiquette x, la probabilité $p(y)$ n'est d'ailleurs pas indispensable. En la considérant comme un facteur constant, nous nous apercevons que la probabilité *a posteriori* est proportionnel à la probabilité jointe des étiquettes et des observations :

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)} \propto p(x)p(y|x) = p(x, y) \tag{2.10}$$

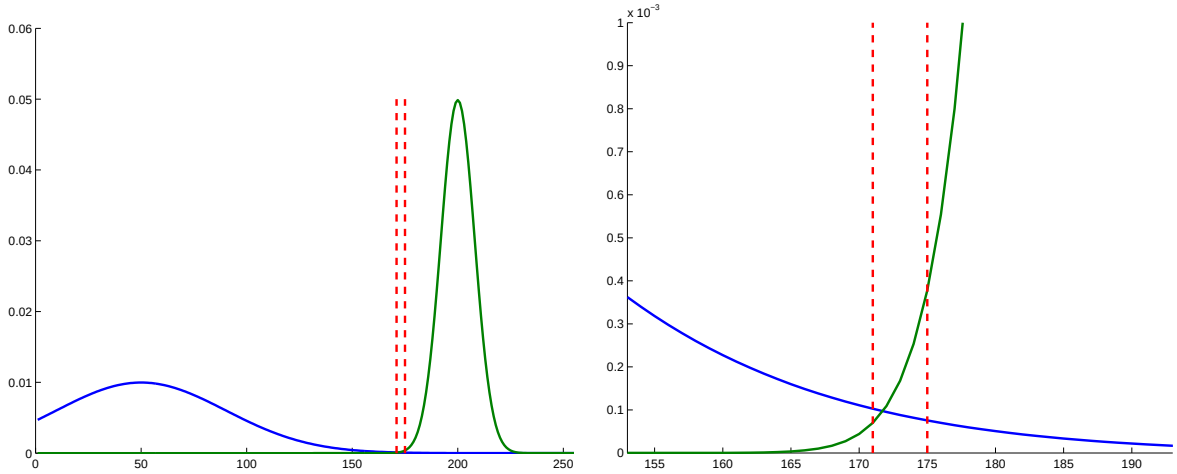
Maximiser la probabilité *a posteriori* est donc équivalent à maximiser la probabilité jointe, ce qui nous emmène au classifier MAP (*maximum a posteriori*) :

$$\hat{x} = \arg\max_x p(x)p(y|x) \tag{2.11}$$

Malgré sa puissance, ce simple *prior* ne fait pas unanimité. Une bien meilleure façon d'injecter une connaissance *a priori* consiste à étudier les relations entre les étiquettes. Ainsi, les décisions ne sont pris de manière indépendante pour chaque pixel, mais sur l'ensemble des pixels en considérant leurs interactions. Afin de gérer la complexité d'un tel modèle, il convient de limiter les interactions gérées. Pour un grand nombre d'applications, une relation de voisinage peut être établie sur l'ensemble des variables. Souvent ces voisinages sont de l'ordre temporel (reconnaissance de parole), spatial et régulier (images), spatial et irrégulier (maillages), spatial uni-dimensionnel (reconnaissance d'écriture) ou spatio-temporel (vidéos). Une hypothèse raisonnable et réaliste consiste à limiter les interactions aux variables prochees, souvent restreint aux voisins directs. D'un point de

vue formel, on considère souvent une dépendance entre toute paire de pixel et une indépendance conditionnelle d'un pixel de tous les autres pixels de l'image sachant ses réalisation de ses voisins. Cette condition de *Markovianité* sera étudiée plus formellement dans le chapitre 3.

Dans notre cas d'application, supposons deux pixels voisins d'une même image, nommés pixel 2 et pixel 3 (c.f. la figure 2.5) avec les niveaux de gris respectifs $y_1 = 171$ et $y_2 = 175$. Nous pouvons associer deux variables X_1 et X_2 aux étiquettes de classes ainsi que deux variables observées Y_1 et Y_2 aux niveaux de gris correspondants. Cela est illustré dans la figure 2.6. A partir de la figure 2.6b nous voyons facilement que la classification par maximisation de la vraisemblance obtiendra $X_1 = 0$ et $X_2 = 1$. La probabilité d'erreur de ce résultat est toutefois probablement assez large car les deux échantillons se situent autour de la zone de l'erreur de Bayes, la zone autour de la valeur y pour laquelle $p(y|X{=}0) = p(y|X{=}1)$. Cela est confirmé par la figure 2.5 — les deux échantillons font effectivement partie de la classe C_1, donc du fond.



(a) (b)

Figure 2.6 – Classification de deux échantillons dépendants — (b) un agrandissement de la région des deux échantillons.

Pour améliorer le modèle, nous pouvons raisonnablement imaginer que deux pixels appartiennent probablement à la même classe, ce qui est justifié par leur proximité. Cette connaissance peut se formaliser en posant un prior sur la probabilité jointe des deux étiquettes :

$$p(x_1, x_2) = \frac{1}{Z} \exp \left\{ \begin{array}{ll} \gamma & \text{si } x_1 = x_2 \\ -\gamma & \text{sinon} \end{array} \right\} \tag{2.12}$$

où Z est un facteur de normalisation nécessaire pour que la somme de probabilités sur toutes les réalisations soit égale à 1. L'exponentielle dans l'équation (2.12) est une façon habituelle de modéliser ce genre d'interactions, qui permet d'assurer la positivité des probabilités toute en gardant une grande flexibilité dans les fonctionnels d'interactions. Nous y reviendrons dans le chapitre 3 sur les modèles graphiques probabilistes. Le paramètre γ contrôle ici la force d'interaction. Une valeur de $\gamma = 0$ annule toute interaction et donc le prior. Une valeur plus forte favorisera d'avantage l'égalité des deux étiquettes X_1 et X_2 dans le cas d'une classification MAP :

$$p(x_1, x_2|y_1, y_2) = \frac{p(y_1|x_1)p(y_2|x_2)p(x_1, x_2)}{\sum_{x'_1, x'_2} p(y_1|x'_1)p(y_2|x'_2)p(x'_1, x'_2)} \tag{2.13}$$

Ici le dénominateur sommera toutes les combinaisons possibles de valeurs pour X_1 et X_2, soit quatre dans cet exemple simple. Nous nous sommes servi d'une hypothèse assez courante, à savoir l'indépendance conditionnelle des observations Y_i sachant les étiquettes X_i :

$$\forall i \in \{1, 2\} : \; p(y_i|x_1, x_2) = p(y_i|x_i) \tag{2.14}$$

Le résultat concret de classification dépend du paramètre d'interaction γ. Si γ est très grand, c.à.d. si une inégalité d'étiquettes est supposée très peu probable (avec une probabilité proche de zéro), le résultat de classification peut être interprété à partir de la figure 2.6b : les deux échantillons seraient alors associés à la classe C_1, ce qui est correct.

A partir de quelle force d'interaction le résultat basculera-t-il en faveur de la solution correcte ? La figure 2.7 montre les probabilités *a posteriori* $p(x_1, x_2|y_1, y_2)$ des quatre résultats possibles

$$\begin{array}{ll} X_1{=}0 & X_2{=}0 \\ X_1{=}0 & X_2{=}1 \\ X_1{=}1 & X_2{=}0 \\ X_1{=}1 & X_2{=}1 \end{array} \tag{2.15}$$

en fonction de la valeur de γ. Les courbes permettent de confirmer les résultats déjà obtenus : à $\gamma = 0$ nous obtenons $X_1 = 0$ et $X_2 = 1$, à une forte valeur nos obtenons $X_1 = X_2 = 1$. Nous pouvons également remarquer que le seuil qui fera basculer le résultat se situe autour de $\gamma = 0.2$.

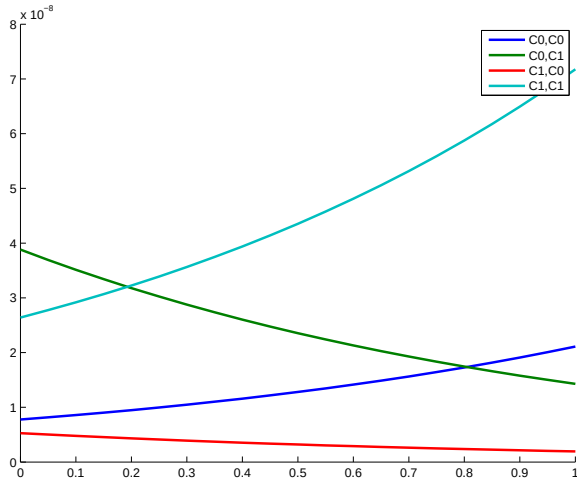


Figure 2.7 – L'effet du paramètre γ sur le résultat de la classification.

L'exemple illustré ici fait intervenir deux échantillons seulement, soit deux pixels voisins. Dans le cas d'une image, la connaissance *a priori* établie un modèle sur tous les pixels de l'image en gérant leurs interactions. Dans ce cas, le gain en performance de classification peut être significatif. Par contre, selon la définition des interactions, l'estimation des étiquettes peut s'avérer assez complexe. Dans le cas de l'exemple ci-dessus, la maximisation de l'équation 2.13 peut être effectuée en calculant toutes les possibles combinaisons de valeurs pour X_1 et X_2, à savoir 4. Dans le cas général, ce problème est de complexité exponentielle, même si, pour certains fonctionnels d'interactions, des solutions polynomiales existent (voir le chapitre 3, section 3.4).

Les problématiques sont très similaires pour les deux exemples décrits ci-dessus. Des décisions doivent être prises sur chaque individu d'un ensemble. Dans les deux cas, les décisions peuvent

être prises de manière individuelle, une par une. Par contre, un gain significatif peut être obtenu en modélisant les interdépendances entre les décisions et en cherchant un résultat optimisant la cohérence globale de la solution, c.à.d. la solution qui respecte au mieux ces interdépendances.

2.3 Organisation du document

Le reste de ce mémoire est organisé comme suit :

Le chapitre 3 décrit la modélisation globalement cohérente par minimisation de fonctions d'énergie. Une introduction assez succincte est donnée sur les modèles graphiques probabilistes (chaînes de Markov, champs de Markov, réseaux Bayesiens) et sur quelques algorithmes de minimisation fréquemment employés. Les exemples simples introduits dans le chapitre 2 seront modélisés de manière plus formelle.

Les chapitres suivants présentent nos contributions originales dans le cadre d'applications divers :

Le chapitre 4 traite la modélisation d'images et de vidéos par modèles structurés et semi-structurés issus d'un représentation par primitives locales. Les graphes, l'appariement de graphes et l'appariement par graphes prennent une place importante dans nos travaux. Ils seront appliqués aux problèmes de la reconnaissance d'objets et la reconnaissance d'activités dans les vidéos.

Le chapitre 5 traite la segmentation et la restauration d'images à l'aide de modèles graphiques probabilistes. Nous nous focaliserons sur la création de modèles tenant compte d'autant d'informations que possible sur la nature de l'image et sur le processus de dégradation pour obtenir un résultat fidèle. L'inférence efficace de la solution sera un thème récurrent dans ce chapitre.

Le chapitre 6 présentera nos travaux sur l'analyse et sur le traitement de modèles géométriques sous forme de maillages surfaciques. Toujours dans le cadre d'une modélisation Markovienne, nos modèles intègrent des informations de nature géométrique et structurelle, tout en modélisant des variables associées à des primitives différentes, telles que les faces, les sommets et les arêtes d'un maillage.

Le chapitre 7 présente les travaux en cours et les nombreuses perspectives. Nous y formulerons une conclusion générale.

Chapitre 3

Les modèles graphiques probabilistes

Ce chapitre présentera, de manière succincte, quelques bases de la modélisation par modèles graphiques probabilistes. L'objectif est de fournir un point de départ pour les travaux développés dans ce mémoire. Quelques familles de modèles seront brièvement présentées, tel que les réseaux Bayesiens (BN), les champs aléatoire de Markov (MRF), les champs aléatoires conditionnels de Markov (CRF), les modèles de Markov cachés (HMM) etc. ; ils seront mis en relation avec nos contributions dans les chapitres suivants. La figure 3.15 sur la page 68 servira comme repère comparant les familles classiques entre elles. Ce chapitre est surtout de nature pédagogique ; les lecteurs travaillant dans les domaines de la vision par ordinateur ou sur l'apprentissage statistique seront très probablement familiers avec la majorité des connaissances présentées.

Ce chapitre est organisé comme suit :

— la section 3.1 donnera une introduction succincte aux aspects théoriques des modèles graphiques probabilistes, sans cibler une application particulière ;
— les sections 3.2 et 3.3 présenteront deux manières différentes de lier les informations observées d'un problème aux variables inconnues, à savoir les modèles génératifs et les modèles discriminatifs ;
— la section 3.4 traitera la minimisation d'énergies associées à ces modèles ;
— la section 3.5 donnera un résumé de l'apprentissage des paramètres ;
— nous conclurons ce chapitre avec une comparaison pédagogique de quelques modèles connus de la littérature, présentée dans la section 3.6, et d'une description des modèles que nous avons développés, présentée dans la section 3.7.

3.1 Modèles probabilistes graphiques

La plupart des problèmes abordés dans ce mémoire partagent une propriété commune : ils peuvent être caractérisés par un ensemble de variables, dont la plupart sont interdépendantes d'une manière ou d'une autre. Les champs aléatoires constituent une approche de modélisation commode pour ce genre de problèmes.

3.1.1 Champs aléatoires

Un champ aléatoire est un ensemble X de variables $X = \{X_1, X_2, \ldots, X_N\}$, chacune pouvant prendre des valeurs dans un ensemble (discret ou continu) $\mathcal{L}_i$ appelé le *domaine*. Les éléments d'un domaine sont appelés *étiquettes*. En pratique, toutes les variables d'un champ partagent souvent le même domaine.

Chaque variable X_i pouvant prendre une valeur parmi les étiquettes admissibles, un ensemble d'étiquettes $x = \{x_1, x_2, \ldots, x_N\}$, une par variable, est nommé une *réalisation*. L'espace de toutes les configurations possibles d'un champ aléatoire sera dénoté $\mathbb{L}$.

> **!** Comme introduit dans le chapitre précédent, nous noterons les variables par des symboles majuscules et les réalisations par des symboles minuscules.

A titre d'exemple nous pouvons citer un champ aléatoire dont les différentes variables correspondent aux différents pixels d'une image. Si le processus aléatoire en question modélise la création d'une image en niveaux de gris, les variables partageront alors le même domaine $\mathcal{L} = \{0 \ldots 255\}$. Si le processus modélise une image segmentée en C classes, alors le domaine unique serait $\mathcal{L} = \{1 \ldots C\}$.

Dans la pratique, on s'intéresse le plus souvent aux probabilités des évènements individuels et collectifs : quelle est la probabilité $p(X_i = x_i)$, c.à.d. la probabilité que la variable X_i prenne la valeur x_i ; quelle est la probabilité jointe $p(X = x)$, ç.à.d. la probabilité que l'ensemble X des variables du champ prenne l'ensemble des étiquette (la réalisation) x ? Dans la suite, nous nous servirons des abréviations $p(x_i)$ pour $p(X_i = x_i)$ et $p(x)$ pour $p(X = x)$, respectivement, pour ce genre de probabilités.

Comment ces deux types de probabilités sont-ils liés ? Comment peut-on obtenir des probabilités inconnues à partir des probabilités connues ou mesurées ? Ces questions sont triviales si les variables sont indépendantes. Dans ce cas

$$p(x) = \prod_i p(x_i).$$

Hélas, les cas les plus intéressants impliquent des variables dépendantes d'une partie d'autres variables. En cas de dépendance, on peut différencier entre dépendance directe, et indépendance conditionnelle. Nous pouvons illustrer cela par un exemple concret impliquant l'acquisition d'une image suivie par sa segmentation par couleurs (niveaux de gris) en C classes. Nous considérons trois variables X_1, X_2 et X_3 correspondant à des mesures différentes sur le même pixel de l'image. La variable X_1 correspond à la quantité de lumière tombée sur la surface correspondant au pixel en question ; la variable X_2 correspond au niveau de gris mesuré pour ce pixel par le capteur CCD de la caméra ; la variable X_3 finalement correspond à la classe à laquelle ce pixel a été associé après la segmentation.

Il est facile de voir que toutes les variables sont inter-dépendantes, c.à.d. aucune variable n'est indépendante d'une autre variable. Par contre, le couple (X_1, X_3) mérite une réflexion supplémentaire. Il est évident que les variables sont dépendantes, puisque le résultat de la segmentation va dépendre de la quantité de lumière du pixel en question. Par contre, le résultat de la segmentation dépendra aussi du niveau de gris du pixel, et cela de façon plus directe. Si nous connaissons le niveau de gris du pixel, c.à.d. la valeur de la variable X_2, alors la connaissance de la quantité de lumière n'apportera plus aucune information supplémentaire. Nous parlerons donc d'une

indépendance conditionnelle entre X_1 et X_3 sachant X_2, notée de la manière suivante :

$$X_1 \perp X_3 \mid X_2$$

ou

$$\begin{aligned} p(x_1|x_2, x_3) &= p(x_1|x_2) \quad \text{et} \\ p(x_3|x_2, x_1) &= p(x_3|x_2) \end{aligned}$$

3.1.2 Modèles graphiques

Dans le reste de ce chapitre nous présenterons différents types de modèles graphiques probabilistes, une famille de modèles pratiques pour gérer les indépendances conditionnelles d'un champ aléatoire. Ces modèles définissent une distribution statistique sur un champ aléatoire à partir d'un graphe associé. De manière générale, les variables du champ sont associées aux sommets (nœuds, sites) du graphe, la topologie (ou structure) du graphe définissant les propriétés d'indépendance (conditionnelle) des variables, c.à.d. la façon par laquelle cette dernière se factorise. Dans ce mémoire nous utiliserons de manière identique les termes « sommet », « nœud » et « site ».

Nous nous servirons de la notation suivante pour décrire les graphes et leurs propriétés : un graphe $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ est caractérisé par un ensemble $\mathcal{V}$ de sommets et un ensemble $\mathcal{E}$ d'arêtes entre les sommets. Un graphe peut être orienté ou non-orienté. Les arêtes du graphe induisent un voisinage entre sommets. Le voisinage N_i d'un sommet i est l'ensemble de tous les sommets avec lesquels il partage des arêtes : $N_i = \{j : (i, j) \in \mathcal{E}\}$. Cette propriété est symétrique, c.à.d. que $i \in N_j \iff j \in N_i$. Deux variables sont voisines si et seulement si $i \in N_j$ et $j \in N_i$.

A chaque sommet est associée une variable aléatoire. Pour faciliter la notation, souvent l'ensemble X des variables est indexé : $X = \{X_1, X_2, \ldots X_N\}$. Cependant, dans certains cas fréquents il est utile d'associer deux ou plusieurs variables au même index. Dans le cas de la modélisation d'un processus aléatoire définis sur une image, un index est souvent associé à un pixel de l'image. Par conséquent, les différentes variables et mesures pour ce pixels seront dénotées par des symboles différents ayant le même indice, e.g. X_i, Y_i etc.

Les deux familles les plus répondues sont les modèles sur graphes orientés, aussi appelé réseaux Bayesiens, et les modèles définis sur graphes non-orientés, aussi appelés champs aléatoires de Markov, ou *Markov Random Fields* (MRF).

3.1.3 Réseaux Bayesiens

Un réseau Bayesien (BN, *Bayesian Network* ou *Belief Network*) est défini sur un graphe orienté sans cycles (en tenant compte des directions des arêtes). Il sert souvent pour modéliser des relations causales entre les variables. La structure du graphe étant orientée, à chaque variable X_i est associé un ensemble de variables parents, que nous dénoterons ici $\mathrm{parents}(X_i)$. La propriété principale d'un BN est la factorisation de la distribution jointe de toutes les variables à partir de la structure du graphe comme suit :

$$p(x_1, x_2, \ldots, x_N) = \prod_i p(x_i|\mathrm{parents}(x_i)) \tag{3.1}$$

Un exemple est donné dans la figure 3.1a illustrant un BN impliquant les variables X_1, X_2 et X_3. La distribution jointe sur l'ensemble de variables se factorise donc comme suit :

$$p(x_1, x_2, x_3) = p(x_1)p(x_2)p(x_3|x_1, x_2) \tag{3.2}$$

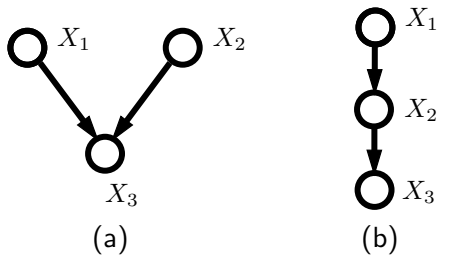
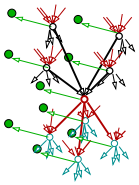


Figure 3.1 – Deux réseaux Bayesiens simples sur 3 variables aléatoires.

Les indépendances conditionnelles des variables d'un BN peuvent se déduire directement de la structure du graphe sous-jacent, raison pour laquelle il est parfois nommé *graphe de dépendances*. Le cas général s'appuie sur un concept nommé *d-séparation* que nous ne détaillerons pas ici [Pea88]. Remarquons seulement que dans le BN illustré dans la figure 3.1b, les variables X_1 et X_3 sont conditionnellement indépendantes sachant la variable X_2. En effet, le BN montré dans la figure 3.1b correspond à l'exemple d'un problème de segmentation d'image donné dans la section 3.1.1. La probabilité jointe des trois variables se factorise de la manière suivante :

$$p(x_1, x_2, x_3) = p(x_1)p(x_2|x_1)p(x_3|x_2) \tag{3.3}$$

La distribution jointe d'un BN n'est pas entièrement définie par le graphe, puisque les différentes facteurs, c.à.d. les probabilités conditionnelles pour chaque variable sachant ses parents, ne peuvent pas être déduites de la structure. Cela fait partie d'une modélisation complémentaire. Dans le cas de notre exemple, il s'agit de définir la distribution $p(x_1)$ et les familles de distributions conditionnelles $p(x_2|x_1)$ et $p(x_3|x_2)$. Dans le cas où le modèle simule la création d'une mesure dans un processus physique, ces informations peuvent être déduites des connaissances sur le processus en question. Dans notre cas, la distribution $p(x_2|x_1)$ pourrait se modéliser par un bruit Gaussian sur le capteur de la caméra. La distribution $p(x_3|x_2)$ dépendra de l'algorithme de segmentation. La distribution $p(x_1)$ dépendra de nos connaissances *a priori* sur l'éclairage de la scène, elle peut être modélisée par une distribution uniforme en l'absence d'informations supplémentaires.



Dans ce mémoire nous nous servons du formalisme des réseaux Bayesiens pour introduire un nouveau modèle servant à la segmentation d'images (c.f. chapitre 5, section 5.4). Plus particulièrement, il modélise les interactions entre des entités différentes associées à une image : pixels et régions de pixels de tailles différentes. Cela permet d'injecter des informations *a priori* connues sur des échelles différentes de l'image.

3.1.4 Champs de Markov aléatoires

Les *champs aléatoires de Markov* (MRF, *Markov Random Fields*) sont des modèles graphiques probabilistes définis sur un graphe non-orienté. Il s'agit donc de modèles non causaux :

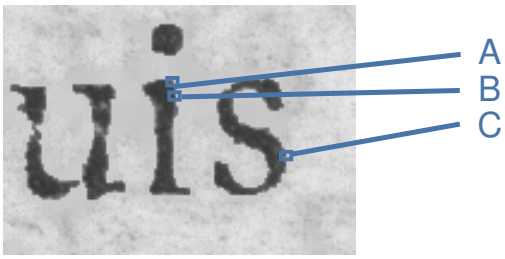


Figure 3.2 – Si les couvertures de Markov pouvaient être déterminées de manière arbitraire à partir des exigences applicatives, ici la couverture de Markov du pixel A contiendrait probablement le site B (faisant partie du même caractère dans l'image), mais pas le site C, sauf si des connaissances supplémentaires de type linguistique étaient intégrées.

Définition 1 *Un champ aléatoire X défini sur un graphe $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ est un* **champ aléatoire de Markov**, *si et seulement si :*

— $p(x) > 0$, $\forall x \in \mathbb{L}$ *: positivité ;*

— $p\left(x_i | x_{\mathcal{S}\setminus\{i\}}\right) = p\left(x_i | x_{N_i}\right)$ *: Markovianité.*

où $\mathcal{S} \setminus \{i\}$ désigne l'ensemble des sommets du graphe sauf le sommet i.

La propriété principale d'un MRF est la Markovianité, qui décrit les indépendances conditionnelles des variables aléatoires. Sans cette propriété, une variable aléatoire peut être potentiellement et directement dépendante de chaque autre variable du champ. La Markovianité rend le processus aléatoire *local* : une variable aléatoire est conditionnellement indépendante des autres variables aléatoires sachant son voisinage (la *couverture de Markov*, ou *Markov blanket*).

Illustrons ce concept par une application, en l'occurrence la binarisation d'une image de document, comme présentée dans la figure 3.2. A chaque pixel est associée une variable aléatoire prenant des valeurs dans $\{0, 1\}$ (*fond, texte*). Si la couverture de Markov pouvait être déterminée de manière arbitraire à partir des exigences applicatives, les voisins de chaque pixel seraient probablement tous les pixels dans un rayon correspondant à la taille d'un caractère. Théoriquement cela pourrait permettre de modéliser la restauration du texte en tenant compte des informations *a priori* sur la forme des caractères. Dans la figure 3.2, la couverture de Markov du pixel A contiendrait le site B (faisant partie du même caractère dans l'image), mais pas le site C, sauf si des connaissances supplémentaires de type linguistique étaient intégrées. Dans la pratique, la taille du voisinage d'un site est surtout limitée par la complexité algorithmique des méthodes de minimisation, ce qui restreint les voisins directs d'un site à un très petit nombre.

Les MRF sont des modèles graphiques probabilistes. Comme pour les réseaux Bayesiens, une distribution de probabilité est donc définie sur l'ensemble des réalisations du champ. Ces distributions sont liées aux propriétés du MRF par le théorème de *Hammersley-Clifford* [HC68, Bes74] :

Théorème 1 *La distribution jointe d'un MRF peut toujours être exprimée comme une fonction de Gibbs :*

$$p(x) = \frac{1}{Z} \exp\left\{ -\sum_c E_c(x_c) \right\} \tag{3.4}$$

où $E(x) = \sum_{c \in \mathcal{C}} E_c(x_c)$ est une fonction d'énergie définie sur les variables des cliques maximales $\mathcal{C}$, et $Z = \sum_x \exp\left\{-\sum_c E_c(x_c)\right\}$ est un facteur de normalisation appelé la fonction de partition.

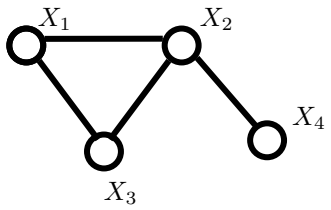


Figure 3.3 – Un MRF simple sur 4 variables impliquant des cliques de taille 1 à 3.

Inversement, chaque distribution de Gibbs correspond à la distribution jointe d'un MRF avec la même structure.

L'équivalence ainsi établie entre les propriétés d'indépendances conditionnelles d'un MRF et la décomposition de sa probabilité jointe est importante car elle permet de spécifier la distribution d'un MRF à partir d'éléments simples, à savoir les potentiels d'énergie $E_c(x_c)$ sur les cliques c. Rappelons ici qu'une clique d'un graphe est un sous-graphe complètement connecté, c.à.d. un sous-graphe pour lequel chaque sommet est lié à chaque autre sommet. Les potentiels d'énergie permettent de favoriser certains configurations (réalisations) du MRF par rapport à d'autre, et donc de modéliser la distribution jointe. Dans ce mémoire nous utiliserons de manière identique les termes « potentiel d'énergie », « potentiel », « fonctionnelle d'énergie » et « fonctionnelle ».

Contrairement aux facteurs d'un BN, qui sont des probabilités conditionnelles, les potentiels d'énergie d'un MRF ne sont pas forcément des probabilités. Au contraire, ils peuvent être définis de manière arbitraire à partir des exigences et des connaissances applicatives. La normalisation de la probabilité jointe est assurée par la fonction de partition Z, garantissant que $0 \leq p(x) \leq 1$.

Un exemple d'un MRF très simple est donné dans la figure 3.3. Le graphe est composé d'une clique de taille trois, de quatre cliques de taille 2, et de 4 cliques de taille 1 (impliquant seulement un sommet). L'énergie du MRF se décompose donc de manière suivante :

$$\begin{aligned} E(x) &= E_1(x_1) + E_2(x_2) + E_3(x_3) + E_4(x_4) + \\ &+ E_5(x_1, x_2) + E_6(x_1, x_3) + E_7(x_2, x_3) + E_8(x_2, x_4) + \\ &+ E_9(x_1, x_2, x_3) \end{aligned} \tag{3.5}$$

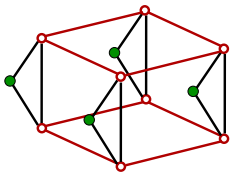
Ici, les E_x sont des fonctions (fonctionnelles) différentes qui peuvent être définies de manière arbitraire. En traitement d'images les cas le plus fréquents concernent les MRF homogènes, où les probabilités conditionnelles et les potentiels d'énergie sont indépendants de la position d'un sommet dans le graphe. L'énergie suivante décrit un MRF homogène pour le graphe dans la figure 3.3 :

$$\begin{aligned} E(x) &= E_1(x_1) + E_1(x_2) + E_1(x_3) + E_1(x_4) + \\ &+ E_2(x_1, x_2) + E_2(x_1, x_3) + E_2(x_2, x_3) + E_2(x_2, x_4) + \\ &+ E_3(x_1, x_2, x_3) \end{aligned} \tag{3.6}$$

ou, reformulé,

$$E(x) = \sum_i E_1(x_i) + \sum_{i \sim j} E_2(x_i, x_j) + \sum_{i \sim j \sim k} E_3(x_i, x_j, x_k) \tag{3.7}$$

ou $i \sim j$ indique deux sommets voisins i et j.



Dans ce mémoire nous nous servons du formalisme des champs aléatoires pour modéliser plusieurs problèmes. Nous introduisons un nouveau modèle servant à la restauration d'images (c.f. chapitre 5, section 5.3). Le modèle gère les interactions entre les pixels de deux pages d'une même feuille d'un document. Dans le cadre de la vidéo, nous proposons un modèle pour l'estimation conjointe du mouvement et la segmentation (voir le chapitre 5, section 5.8). Les modèles de type MRF nous sont également utiles pour la segmentation de maillages 3D en primitives géométriques cohérentes (voir le chapitre 6).

3.1.5 Étapes de la modélisation d'un problème donné

Les applications considérées dans ce mémoire ont recours à la modélisation par un processus aléatoire pour résoudre un problème spécifique. Cela implique généralement l'estimation d'une ou plusieurs variables inconnues (cachées) à partir d'une ou plusieurs valeurs connues, appelées *observations* ou *mesures*. Une convention répandue, que nous adopterons également dans ce mémoire, consiste à dénoter les variables observées de manière différente et de les illustrer de manière différente dans le graphe de dépendances, par exemple par des ronds opaques à l'opposé des ronds creux associés aux variables cachées. Dans la suite de cette section, sauf indiqué autrement, nous adopterons la convention de noter les variables cachées par la lettre X et les variables observées par la lettre Y. Un exemple pour un graphe comprenant des sommets des deux types est donné dans la figure 3.5.

Quelque soit le type de modèle choisi, modèle sur graphe orienté (BN) ou modèle sur graphe non-orienté (MRF), de manière générale la solution d'un problème donné par modèles graphiques probabilistes implique plusieurs étapes :

Modélisation du graphe La première étape consiste à identifier toutes les variables, cachées et observées, à identifier les indépendances conditionnelles et à construire le graphe de dépendances. Cela va déterminer la factorisation de la probabilité jointe des variables, et donc la décomposition de la fonction d'énergie en termes. Dans l'exemple du graphe donné dans la figure 3.3, la fonction d'énergie (3.7) est obtenue pour un MRF homogène.

Le terme « fonction d'énergie », venant des modèles MRF, n'est pas employé dans la littérature sur les réseaux Bayesiens. Pour simplifier, nous nous servirons abusivement du terme « fonction d'énergie » pour le négatif du logarithme de la probabilité jointe d'un BN, $-\ln p(x, y)$. Cela est également motivé par le fait que les calculs d'inférence d'un BN sont fait dans le domaine logarithmique afin d'éviter des instabilités numériques.

Modélisation de la fonction d'énergie La forme fonctionnelle de l'énergie d'un modèle n'est pas complètement déterminée par le graphe de dépendances. Une étape supplémentaire consiste à spécifier la forme fonctionnelle des potentiels d'énergie sur les cliques. Pour le MRF décrit ci-dessus, il s'agit d'identifier les potentiels $E_1(.)$, $E_2(.,.)$ et $E_3(.,.,.)$.

Estimation des paramètres Hormis les cas les plus simples, la fonction d'énergie, décomposée en potentiels d'énergie sur les cliques, contient habituellement des paramètres. Une étape d'apprentissage va estimer ces paramètres de manière supervisée ou non-supervisée à partir d'un ensemble de données d'apprentissage.

Optimisation/Minimisation Une fois la fonction d'énergie entièrement définie, une probabilité est associée à chaque réalisation d'un MRF. Dans le cas classique, le problème consiste à trouver la réalisation la plus probable des variables cachées étant donnée une partie des valeurs des variables nommées *observations*. Dans ce cas on parle de l'estimateur MAP (*Maximum A Posteriori*) :

$$\hat{x} = \arg\max_x p(x) = \arg\min_x E(x) \tag{3.8}$$

D'autres estimateurs sont possibles.

Quand il s'agit de modélisations probabilistes, par modèle graphique ou non, deux familles de modélisation se sont établies : les modèles *génératifs* et les modèles *discriminatifs*. Chacune de ces deux familles gère les interactions entre variables observées et les variables cachées à sa manière. De manière plus précise, la probabilité postérieure $p(x|y)$, souhaitée afin de pouvoir estimer la réalisation optimale pour la variable x, peut être modélisée de manières différentes. Les modèles génératifs établissent un lien permettant de produire (c.à.d. d'échantillonner) des réalisations pour les observations à partir d'échantillons des variables cachées. Cela est montré dans la figure 3.4b : la probabilité postérieure est décomposée en deux facteurs, la loi *a priori* $p(x)$ et la vraisemblance $p(y|x)$. Cela est pratique quand on sous-entend que la valeur Y est produite à partir de X. Obtenir la loi postérieure nous demande d'inverser le modèle.

En contrepartie, les modèles discriminatifs ne s'intéressent pas à la manière par laquelle sont générées les observations du système. Comme leur nom l'indique, ils modélisent directement le processus d'estimation des variables cachées, donc l'état du système, à partir des informations mesurées. La situation est montrée dans la figure 3.4c : le lien entre X et Y est inversé par rapport au modèle génératif. Cela peut sembler moins intuitif d'un point de vue science expérimentale, puisque le modèle obtenu est plus loin de la réalité supposée. En l'occurrence, les résultats sont souvent meilleurs quand il s'agit de problèmes de classification et de segmentation.

En ce qui concerne les modèles graphiques probabilistes, la différence entre les deux familles intervient à plusieurs niveaux :

- Les modèles génératifs modélisent la probabilité jointe $p(x, y)$. La probabilité marginale $p(y)$ fait donc partie du modèle. Par contre, les modèles discriminatifs modélisent la probabilité conditionnelle $p(x|y)$. Les paramètres du modèle ne sont pas utilisés pour la modélisation de $p(y)$, inutile pour la classification.
- L'inversion du modèle nécessaire pour les modèles génératifs se fait habituellement dans un cadre Bayesien en appliquant la règle de Bayes. Toutefois, l'étape d'estimation de paramètres après inversion peut s'avérer complexe pour des graphes arbitraires. En conséquence, les structures de graphes utilisées pour les modèles génératifs sont souvent plus restreintes que celles pour les modèles discriminatifs.
- La différence principale entre les deux familles concerne les algorithmes d'apprentissage. En effet, si l'apprentissage de paramètres n'est pas demandé, par exemple en fixant tous les paramètres de la fonction d'énergie de manière manuelle, aucune différence entres les deux familles de modèles ne peut être observée, si on fait abstraction d'une éventuelle différence d'interprétation purement philosophique des potentiels d'énergie.

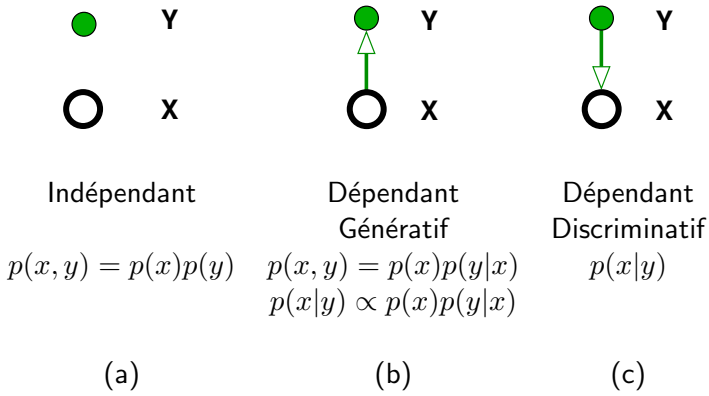


Figure 3.4 – Possibilités d'interactions entre des variables observées et des variables cachées : (a) indépendance ; (b) un modèle génératif ; (c) un modèle discriminatif.

Dans les deux sections suivantes (3.2 et 3.3) nous donnerons plus de détails sur ces deux familles de modèles et nous donnerons quelques exemples.

3.2 Modèles génératifs

Dans le cas des modèles génératifs, on conjecture souvent l'existence, dans la nature, de variables inconnues décrivant l'état du système. On modélise également le processus produisant des échantillons des mesures à partir d'une réalisation des variables cachées. Cette façon de modéliser un problème convient parfaitement pour un grand nombre de processus physiques bien connus et/ou étudiés.

Dans un contexte Bayesien, les connaissances du système en question sont séparées en deux parties

- la connaissance *a priori* de l'état du système, le *prior*, noté $p(x)$. A chaque réalisation de l'ensemble des variables cachées est associée une probabilité. On ne tient pas compte des observations ;
- le modèle d'observation, aussi nommé la vraisemblance des observations, noté $p(y|x)$. Cette partie du modèle établit le lien causal entre les deux types de variables.

L'inférence des variables cachées à partir des variables observées peut se faire en maximisant la probabilité *a posteriori* $p(y|x)$. Étant donné le *prior* et la vraisemblance des données, elle s'obtient en appliquant la règle de Bayes :

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)} \tag{3.9}$$

L'estimateur MAP maximise cette probabilité pour obtenir la réalisation la plus probable des variables cachées :

$$\begin{aligned} \hat{x} &= \arg\max_x p(x|y) \\ &= \arg\max_x \frac{p(x)p(y|x)}{p(y)} \\ &= \arg\max_x p(x)p(y|x) \\ &= \arg\max_x p(x,y) \end{aligned} \tag{3.10}$$

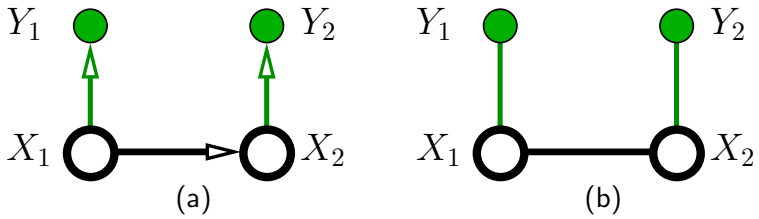


Figure 3.5 – Deux modèles graphiques possibles modélisant l'exemple de segmentation décrite dans la section 2.2. Les sommets opaques en vert correspondent aux observations, les sommets creux correspondent aux variables cachées. (a) une version réalisée comme un réseau Bayesien ; (b) une version réalisée comme un MRF.

Cela revient donc à maximiser la probabilité jointe des variables. Pour illustrer cela, revenons pour un instant à l'exemple de segmentation d'images décrit dans la section 2.2. L'idée était de modéliser la dépendance des résultats de segmentation de deux pixels voisins par un modèle Bayesien impliquant un *prior* ainsi qu'un facteur de vraisemblance des données observées. Ce modèle, défini sur les variables cachées X_1 et X_2 (les étiquettes) et les variables observées Y_1 et Y_2 (les niveaux de gris), décrit par les équations (2.12)-(2.13), peut être présenté par le réseau Bayésien montré dans la figure 3.5a.

Le graphe de dépendances donne la distribution jointe suivante :

$$p(x_1, x_2, y_1, y_2) = p(x_1)p(x_2|x_1)p(y_1|x_1)p(y_2|x_2) \tag{3.11}$$

Pour arriver au modèle décrit dans (2.12)-(2.13), il suffit de définir les probabilités conditionnelles comme suit :

— Les facteurs de vraisemblance sont définis comme dans (2.5)

— le modèle à priori est défini par

$$p(x_1) = 0.5 \tag{3.12}$$

$$p(x2|x1) = \frac{1}{Z}\exp\left\{\begin{array}{cc} \gamma & \text{si } x_1 = x_2 \\ -\gamma & \text{sinon} \end{array}\right\} \tag{3.13}$$

où $Z = \exp(\gamma) + \exp(-\gamma)$.

La paramètre γ contrôlant la force de régulation est celui défini dans (2.12), page 32.

L'exemple ci-dessus peut également être réalisé avec un modèle non-orienté, c.à.d. un MRF. Le graphe de dépendances est similaire au graphe du BN, il suffit de transformer les flèches en arêtes non-orientées (voir la figure 3.5b). La probabilité jointe des variables s'exprime alors comme une distribution de Gibbs :

$$p(x_1, x_2, y_1, y_2) = \frac{1}{Z}\exp\left\{-(E_1(x_1, y_1) + E_1(x_2, y_2) + E_2(x_1, x_2)\right\} \tag{3.14}$$

Les potentiels sont donnés comme le logarithme des probabilités conditionnelles de la version réseaux Bayesien multiplié par un facteur -1 :

$$E_1(x_i, y_i) = -\ln\left\{\begin{array}{ll} \mathcal{N}(y_i; \mu_0, \sigma_0) & \text{si } x = 0 \\ \mathcal{N}(y_i; \mu_1, \sigma_1) & \text{si } x = 1 \end{array}\right.$$

$$E_2(x_i, x_j) = \left\{\begin{array}{cc} -\gamma & \text{si } x_i = x_j \\ \gamma & \text{sinon} \end{array}\right. \tag{3.15}$$

Notons que le facteur de normalisation Z, présent dans (3.13), disparait du prior $E_2(.)$ donné dans (3.15). Il n'est pas nécessaire comme les potentiels d'énergie d'un MRF ne sont pas forcément des probabilités. La normalisation est faite par la fonction de partition Z dans (3.14).

Comme leur nom l'indique, les modèles génératifs permettent de facilement générer (de simuler) des données. Cela se voit particulièrement bien dans la version réseau Bayesien de notre exemple, grâce à la nature causale du modèle. Un raisonnement similaire est toutefois applicable à la version en MRF également.

Pour échantillonner une réalisation (y_1, y_2) des variables observées (Y_1, Y_2), nous procédons dans l'ordre des liens causaux en commençant par la variable X_1. Une réalisation x_1 peut être obtenue en échantillonnant selon la loi (3.12). La valeur x_1 obtenue, une valeur x_2 peut être obtenue en échantillonnant selon (3.13). Les valeurs pour les variables cachées ainsi déterminées, les observations sont échantillonnées selon la vraisemblance des données (2.5).

Dans la suite nous donnerons deux exemples classiques de modèles graphiques probabilistes génératifs.

3.2.1 Les modèles de Markov cachés

Un *modèle de Markov caché* (*Hidden Markov Model, HMM*) permet de décrire un processus dynamique, souvent évoluant dans le temps, c.à.d. le comportement temporel d'une séquence d'observations $\{Y_1, Y_2, \ldots, Y_N\}$. Contrairement à un modèle auto-régressif ou une chaîne de Markov simple, qui modélisent les interactions entre les observations directement, un HMM introduit un état interne inconnu X_t par instant t. Le processus dynamique modélise les interactions par trois types de paramètres :

- l'interaction entre les états successifs est donnée par un ensemble de distributions conditionnelles, les probabilités de transitions $p(x_t|x_{t-1})$;
- la connaissance sur le premier état à l'instant t_0 est donné par la distribution $p(x_0)$;
- les interactions entre les états et les observations sont données par un ensemble de distributions conditionnelles, les probabilités d'émission $p(y_t|x_t)$.

Il est possible de dérouler le modèle dans le temps pour obtenir un graphe de dépendances, illustré dans la figure 3.6a. Il est facile de voir qu'un HMM peut ainsi être vu comme un cas spécial d'un réseau Bayesien avec une structure graphique très spécifique : les variables cachées sont organisées de façon linéaire en une chaine. Les variables observées sont liées chacune à une variable cachée spécifique.

Les applications classiques pour les HMM sont les applications liées aux signaux 1D où l'axe du signal correspond au temps, par exemple la reconnaissance de parole etc. Pour cette raison, le terme *transition* est utilisé pour le passage d'une valeur pour une variable X_t à une autre valeur pour la variable suivante X_{t+1}. Les distributions de probabilités décrivant ces transitions peuvent être illustrées par un graphe appelé le graphe de transitions. Dans ce graphe, dont un exemple est montré dans la figure 3.6b, un sommet correspond à un état possible pour chaque variable X_t, donc à une étiquette des variables cachées. A chaque arête entre deux états est attribuée sa probabilité de transition. Notons que le graphe n'est pas forcement complètement connecté : certaines transitions peuvent être interdites. La figure 3.6c, montre un type de modèle fréquent, les modèles gauche-droite. Dans ces modèles, le graphe de transitions ressemble au graphe de dépendances : les états sont parcourus de manière linéaire, chaque état étant sélectionné une fois, éventuellement pendant plusieurs instants t consécutifs, avant de passer la main à l'état suivant

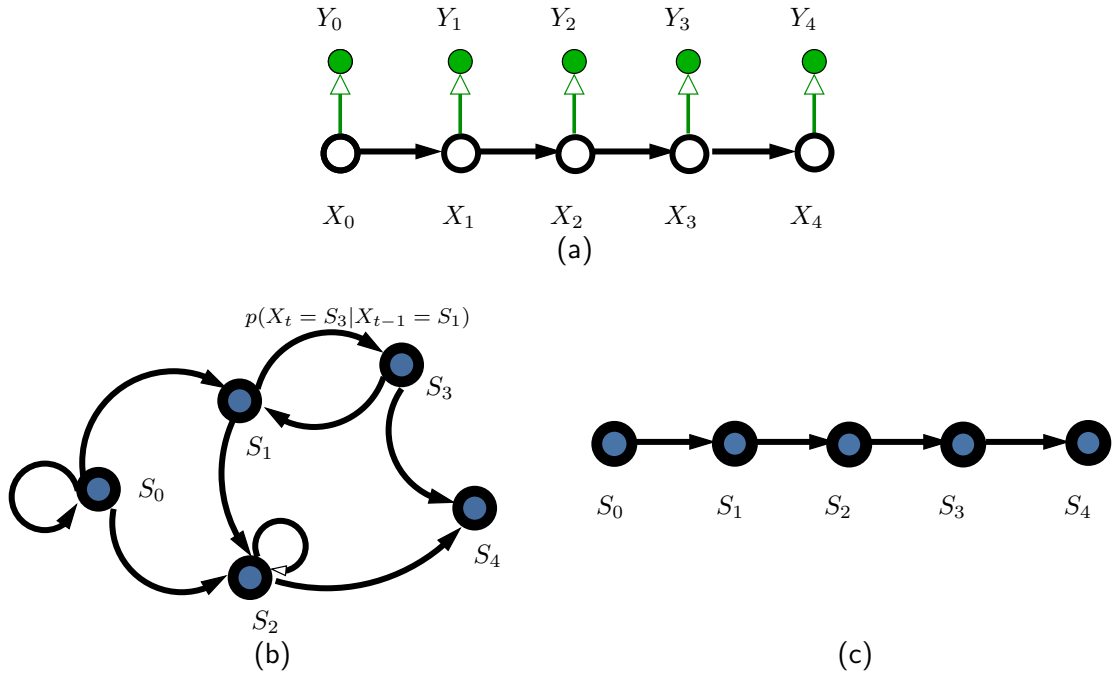


Figure 3.6 – Un HMM : (a) le graphe de dépendances déroulé ; (b) un graphe de transitions ; (c) un graphe de transitions de type gauche-droite.

dans le graphe de transitions. Il est important de noter que les variables du graphe de dépendances ne correspondent pas aux états dans le graphe de transitions.

Revenons maintenant au graphe du dépendances du modèle. Sa structure linéaire est notable pour une propriété importante, à savoir l'absence de cycles, même en ignorant le sens des flèches [1]. Grâce à cette propriété, les algorithmes d'apprentissage et les algorithmes d'inférence du meilleur étiquetage peuvent être formulés de manière efficace. Notamment l'étiquetage peut être résolu de manière exacte en un temps polynomial, à savoir $O(N.|\mathcal{L}|^2)$, N étant la longueur de la chaîne, et $|\mathcal{L}|$ étant le nombre d'étiquettes, c.à.d. d'états. Plus de détails sont données dans la section 3.4.1 de ce chapitre.

Ce modèle a également été appliqué à la segmentation d'images [PT00, FDP+03, SC06], malgré sa structure linéaire. Pour cela, un parcours fractal de type Hilbert-Peano est défini sur les pixels de l'image (cf. la figure 3.7), la chaîne du modèle suivant ce parcours. Bien évidemment le voisinage induit par ce parcours ne peut pas entièrement reproduire le voisinage d'une grille 2D. Le résultats sont néanmoins très proches aux résultats obtenus par un modèle défini sur une grille 2D, par exemple les MRF (cf. la section suivante), tout en gardant l'avantage d'une complexité de calcul très faible [SC06].

1. Rappelons nous que le graphe de dépendances d'un réseau Bayesien ne peut avoir de cycles en tenant compte du sens des flèches.

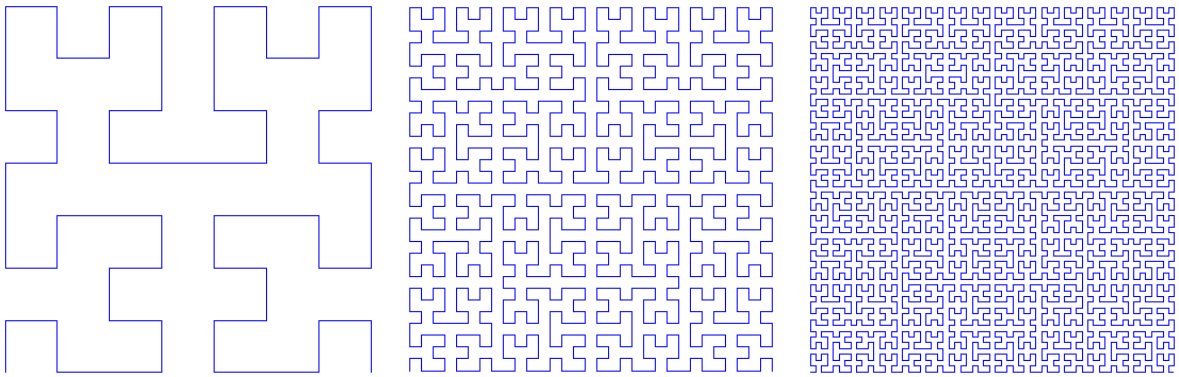


Figure 3.7 – Le parcours fractal Hilbert-Peano d'une grille 2D : quelques itérations.

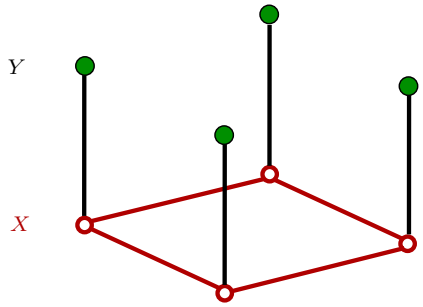


Figure 3.8 – Un HMRF pour une image de taille 2×2 avec un voisinage de type 4-connecté.

3.2.2 Les champs de Markov cachés

Les champs de Markov cachés (*Hidden Markov Random Fields*, HMRF)[2] sont une famille de modèles fréquemment utilisés pour la segmentation et/ou le dé-bruitage (la restauration) d'images. D'autres applications concernent la reconnaissance d'objets spécifiques, souvent combinée avec la segmentation de l'objet. A chaque pixel i sont associées deux variables, une variable cachée X_i et une variable observée Y_i. Le graphe de dépendances est composé de deux types d'arêtes :

- des cliques de taille 2 liant chaque variable cachée X_i à son observation correspondante Y_i. Dans un cadre Bayesien, il s'agit d'arêtes associées aux termes d'attache aux données (la vraisemblance des observations) ;
- des cliques de taille 2, ou plus, liant les variables cachées entre elles. Il s'agit d'arêtes associées aux termes de régularisation, le *prior* dans le cadre Bayesien.

La figure 3.8 montre un exemple d'un graphe de dépendances pour une très petite image de taille 2×2 avec un voisinage de type 4-connecté. Un exemple pour une fonction d'énergie répondue est le modèle de Potts [Pot52, Li01] favorisant des étiquettes identiques pour les variables voisins :

$$E(x) = \sum_i E_d(x_i, y_1) + \mu \sum_{i \sim j} \delta(x_i, x_j) \tag{3.16}$$

2. Dans la littérature, le terme champ de Markov caché est utilisé de manière incohérente pour plusieurs types de modèles. Il peut également dénoter, entre autres, un champ avec deux variables cachées par site.

où $\delta(.,.)$ est le Kronecker delta défini par

$$\delta(a,b) = \begin{cases} 1 & \text{si } a = b \\ 0 & \text{sinon} \end{cases} \tag{3.17}$$

Ici $E_d(.,.)$ est un terme d'attache aux données, par exemple une fonction issue d'un modèle basé sur un bruit Gaussien, et μ est un paramètre pondérant les deux types de potentiels.

La nature non-causale et surtout la présence de cycles dans un graphe, indiquant des dépendances circulaires entre les variables, rendent les algorithmes d'estimation plus complexes que les algorithmes pour les modèles linéaires tels que les HMM. Pour l'inférence des étiquettes à partir d'un estimateur MAP, les algorithmes de minimisation par calcul de coupure minimale dans un graphe (présentés dans la section 3.4.4), récemment redécouvert dans ce contexte, permettent de trouver une solution exacte dans un temps polynomial pour certaines classes de fonctions. Plus de détails sont donnés dans la section 3.4.4 de ce chapitre. Quant à l'estimation de paramètres, aucun algorithme efficace n'est connu donnant une solution exacte du problème. Les algorithmes existants passent souvent par des approximations du modèle ou du calcul de l'estimation même (voir la section 3.5.1).

3.2.3 Restrictions

Les modèles génératifs modélisent la probabilité marginale $p(x)$ (le *prior*) explicitement, une probabilité peut donc être calculée facilement pour une réalisation des variables cachées. Une grande partie de la littérature sur les MRF considère d'ailleurs le champ X comme le MRF proprement dit d'un modèle donné, la vraisemblance $p(y|x)$ étant considérée comme à part. Par contre, la plupart des formes fonctionnelles pour $p(y|x)$ se laissent facilement exprimer comme des termes définis sur des cliques supplémentaires du MRF, ce qui permet l'interprétation de l'ensemble du modèle $p(x,y)$ comme un MRF. La différence est purement philosophique sans effet sur la modélisation ou les algorithmes d'inférence. Pour les deux cas, le *prior* $p(x)$ est explicitement modélisé.

Un inconvénient majeur de cette séparation entre le *prior* et la vraisemblance est l'inversion du modèle nécessaire pour obtenir le *posterior* $p(x|y)$. Pour rendre possible l'inférence, des restrictions fortes sont souvent imposées sur la forme de la vraisemblance $p(y|x)$. Habituellement on impose les restrictions suivantes :

1. Les observations Y_i sont conditionnellement indépendantes sachant les variables cachées X_i
2. $p(Y_i|X) = p(Y_i|X_i)$

(3.18)

Notons que ces restrictions sont valides pour les HMM (cf. la figure 3.6) et les HMRF (cf. la figure 3.8) — voir une explication dans la section 3.5 sur l'inférence des paramètres.

3.3 Modèles discriminatifs

Les modèles discriminatifs modélisent directement l'inférence des variables cachées à partir des observations — les caractéristiques ou *features* en langage apprentissage. Contrairement aux modèles génératifs, leur paramètres ne sont pas utilisés pour décrire la distribution $p(y)$ des observations, inutile dans un contexte de classification où l'on s'intéresse surtout à l'estimation des variables

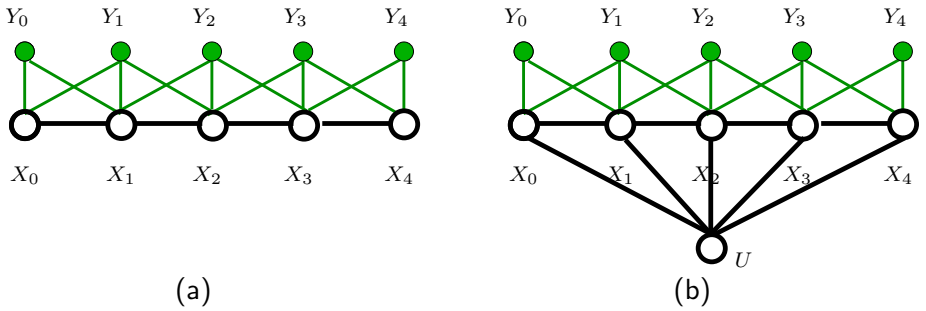


Figure 3.9 – Deux modèles discriminatifs : (a) un CRF à structure linéaire [LMP01] ; (b) un HCRF [QWM+07].

cachées. A la place, la probabilité postérieure $(x|x)$ est modélisée directement. Un exemple non-graphique de modèles discriminatifs opérant sur des données plongées dans un espace vectoriel sont les classifieurs, sont la régression logistique ou les réseaux de neurones (s'ils sont entrainés de manière spécifique). Pour un grand nombre d'applications de classification, les modèles discriminatifs sont plus efficaces que les modèles génératifs, sans pour autant être plus performant pour *tous* les problèmes [NJ02].

Des versions discriminatives pour les modèles structurés comme les HMM, MRF et BN ont été introduites. L'exemple le plus répondu est sans doute le champ aléatoire conditionnel, ou *Conditional Random Field* (CRF) [LMP01] :

Définition 2 *Soit un champ aléatoire X structuré par un graphe $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Soit un champ aléatoire Y. L'ensemble (X, Y) est un champ aléatoire conditionnel si X est un champ aléatoire de Markov quand il est conditionné sur Y, c.à.d. que la condition de Markovianité est satisfaite :*

$$p\left(x_i|y, x_{\mathcal{S}\setminus\{i\}}\right) = p\left(x_i|y, x_{N_i}\right)$$

Contrairement à un MRF classique, la distribution des observations $p(y)$ ne fait pas partie du modèle. Un CRF est conditionné sur les observations, mais il ne les modélise pas. Cela présente plusieurs avantages. Tout d'abord, comme nous l'avons mentionné ci-dessus, les paramètres du modèle ne sont pas utilisés pour la modélisation des observations, inutile pour la classification. Deuxièmement, lors de l'apprentissage d'un modèle génératif, l'estimation de la distribution des observations nécessite que l'espace des observations soit complètement couvert. Cela peut poser des problèmes si certaines réalisations sont très rares. Une estimation inexacte peut en effet faire chuter la performance d'un modèle génératif.

Les CRF étaient initialement introduits avec une structure linéaire pour des séquences de données [LMP01], traditionnellement gérées par des HMM. Dans ce cas, le graphe de dépendances, montré dans la figure 3.9a, peut être obtenu à partir du graphe d'un HMM (montré dans la figure 3.6a) en le convertissant en un graphe non orienté, donc en supprimant les orientations des flèches. Nous pouvons pourtant voir des arêtes supplémentaires dans le graphe du CRF par rapport à la version HMM. En effet, les conditions d'indépendances conditionnelles des observations (3.18), nécessaires pour l'apprentissage des paramètres d'un modèle génératif, peuvent être relâchées pour

les CRF. Cela présente un vrai avantage, une observation pouvant être liée à plusieurs variables cachées, et vice versa.

La structure d'un CRF n'est pas restreinte à une chaine. Les *Hidden Conditional Random Fields*, dont un exemple est illustré dans la figure 3.9b, sont une extension de la structure linéaire pour la classification de séquences [QWM+07]. Dans le graphe, une variable cachée supplémentaire U, liée à toutes les autres variables cachées, modélise la classe de la séquence. Les CRF aux structures générales existent, c.à.d. les CRF ayant des connectivités de graphes non spécifiques. Ils partagent des propriétés similaires avec les MRF à structure générale. La présence potentielle de cycles dans le graphe rend difficile l'inférence des étiquettes et des paramètres — voir les deux sections suivantes. Ils sont néanmoins fréquemment utilisés en vision par ordinateur, par exemple pour la reconnaissance et la segmentation conjointe d'objets [WS06, HRW07].

Les CRF étant des MRF, leurs fonctions d'énergie peuvent se noter sous la même forme (3.4). Pour des raisons historiques — les CRF sont issus de la communauté apprentissage — une notation et une forme spécifique se sont établies, où les potentiels d'énergie sont nommées « fonctions de caractéristiques » (*feature functions*) f_k :

$$p(x|y) = \frac{1}{Z(y)} \exp\left\{ \sum_{c \in \mathcal{C}, k} \lambda_{ck} f_k(x_c, y) \right\} \tag{3.19}$$

Ici, la somme est sur toutes les cliques c du graphe sur X et sur les différentes fonctions pour chaque clique c ; les λ_{ck} sont les paramètres associés aux cliques c et aux fonctions f_k. Notons également que chaque fonction f_k pour une clique c donnée peut potentiellement dépendre de toutes les observations y du modèle. Il est important de noter que la forme (3.19) d'un CRF est moins générale que la forme (3.4) d'un MRF : les fonctions f_{ck} ne peuvent comprendre aucun paramètre supplémentaire.

3.4 Inférence des étiquettes : minimisation d'énergies

L'inférence des étiquettes a comme objectif d'estimer la réalisation la plus probable des variables cachées étant données une réalisation des variables observées. Cela suppose une estimation des paramètres du modèle, décrite dans la section suivante. Ici nous nous concentrations sur les méthodes procédant par la maximisation d'une probabilité ou par la minimisation d'une fonction d'énergie globale, ce qui est équivalent pour les modèles traités dans ce mémoire. Pour les MRF ou CRF, la fonction d'énergie est une partie intégrale du modèle. Pour les modèles sur graphe orienté (réseaux Bayésien, HMM etc.), un équivalent d'une fonction d'énergie peut être obtenu par le négatif du logarithme de la probabilité jointe des variables, s'il s'agit d'un modèle génératif :

$$E(x) = -\log p(x, y) \tag{3.20}$$

ou en passant directement par la probabilité conditionnelle pour les modèles discriminatifs :

$$E(x) = -\log p(x|y) \tag{3.21}$$

Pour aboutir à ces équations, nous avons supposé l'estimateur classique MAP (*Maximum A Posteriori*) :

$$\hat{x} = \arg\max_x p(x|y) \tag{3.22}$$

D'autres estimateurs sont possibles, leur discussion dépasse le sujet de ce mémoire. Il est néanmoins utile de mentionner un autre estimateur assez répondu, le MPM (*Mode of Posterior Marginals*) :

$$\hat{x_i} = \arg\max_{x_i} p(x_i|y) \tag{3.23}$$

Selon la structure graphique du modèle, le calcul de (3.23) peut être plus difficile que le calcul de (3.22), car il nécessite l'intégration sur les autres variables $x_j, j \neq i$. En revanche, pour les problèmes de segmentation, il donne souvent des biens meilleurs résultats.

Le modèle défini et tous les paramètres éventuels estimés, la solution est donc obtenue par une minimisation sur toutes les valeurs possibles du champ caché x :

$$\widehat{x} = \arg\min_{x} E(x) \tag{3.24}$$

Une minimisation « brute » nécessiterait l'énumeration d'un nombre exponentiel de réalisations différentes, à savoir $|\mathcal{L}|^N$, où $\mathcal{L}$ est l'ensemble de valeurs possible pour chaque variable x_i et N est le nombre de variables. Bien évidemment cela est possible seulement pour des problèmes de taille extrêmement petite et donc peu utiles. Le calcul de la solution exacte du problème général est difficile. Les méthodes de minimisation connues se basent sur des propriétés spécifiques d'un modèle :

- pour certaines structures de graphes, par exemple des chaînes ou des arbres, la solution exacte du problème peut être calculée en temps polynomiale ;
- pour certains potentiels, la solution exacte du problème peut être calculée en temps polynomiale, quelque soit la structure du graphe ;
- si la solution exacte ne peut être obtenue de manière efficace, un ensemble de techniques permet d'obtenir un minimum local, parfois avec des garanties de convergence ;
- alternativement, dans certains contextes il peut être plus intéressant d'approximer la structure du graphe pour obtenir une solution exacte du problème approché.

Les sous-sections suivantes donneront quelques pistes.

3.4.1 Programmation dynamique et algorithme de Viterbi

Pour certains modèles, la structure du graphe suit une forme bien spécifique. A titre d'exemple, le graphe de dépendances d'un HMM est un arbre spécifique composé d'une chaîne, formée par les variables cachées, et les variables observées sont chacune liée à une variable cachée (voir la figure 3.6a). Les variables X_i et Y_i suivent donc un ordre strict et leurs indices i en dépendent. En conséquence, la forme de la fonction d'énergie est simplifiée :

$$E(x) = \min_{x} \sum_{i=1}^{N} E_1(x_i, y_i) + \lambda \sum_{i=2}^{N} E_2(x_i, x_{i-1}) \tag{3.25}$$

La structure spécifique permet de réordonner les opérateurs $\min$ et $+$ pour obtenir l'équation suivante, donnée pour le cas de $N{=}3$ pour des raisons de simplicité :

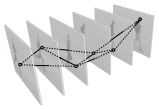
$$\min_{x_1,x_2,x_3} E_1(x_1,y_1) + E_2(x_2,x_1) + E_1(x_2,y_2) + E_2(x_3,x_2) + E_1(x_3,y_3) =$$

$$= \min_{x_1}\Bigg[E_1(x_1,y_1) + \min_{x_2}\Bigg[E_1(x_2,y_2) + E_2(x_2,x_1) + \underbrace{\min_{X_3}\Big[E_1(x_3,y_3) + E_2(x_3,x_2)\Big]}_{R_3(x_2)}\Bigg]\Bigg] \tag{3.26}$$

Cela peut être exploité en définissant la récursion suivante, et en introduisant une variable $R_i(x_{i-1})$:

$$R_i(x_{i-1}) = \min_{x_i}\left[E_1(x_i,y_i) + E_2(x_i,x_{i-1}) + R_{i+1}(x_i)\right] \tag{3.27}$$

Cette récursion permet de calculer la solution exacte avec une complexité de $O\left(N|\mathcal{L}|^2\right)$. Pour plus de détails nous renvoyons le lecteur à [Rab89], où l'algorithme Viterbi [Vit67] est détaillé : une variante de cet algorithme dédiée aux HMM.



Dans ce mémoire nous généralisons cette technique aux graphes dont la structure n'est pas linéaire, mais plus généralement "alongée". Ce type de graphes peut être obtenu à partir de point saillants détectés dans une vidéo en se servant de propriétés de proximité. Plus de détails seront donnés dans le chapitre 4.

3.4.2 L'algorithme max-produit et la propagation de croyances

L'extension du principe de la programmation dynamique (voir la section précédente) aux arbres est connu sous le nom d'algorithme *max-produit* ou *propagation de croyance*, cf. [Pea88, Bis06, KFL01]. Il a été introduit dans un contexte probabiliste et opère donc directement sur la probabilité jointe. A partir de la racine, ou à partir d'un sommet arbitrairement choisi comme étant la racine, la récursion procède en direction des feuilles.

Cet algorithme est aussi souvent appliqué de manière itérative pour obtenir une solution approchée du problème pour un graphe de structure arbitraire, donc contenant des cycles (*Loopy Belief Propagation*). Aucune garantie de convergence n'est connue, à l'exception de quelques cas spécifiques avec des structures graphiques très simples [Wei00]. En pratique, l'algorithme converge pour un grand nombre de structures graphiques [WF01, YFW05]. Une étude empirique indique que les distributions marginales obtenues par la propagation de croyances convergent souvent vers des bonnes approximations des distributions marginales *a posteriori* [MWJ99]. Dans un contexte de codage d'un canal d'informations, il a été montré que l'algorithme efficace *Turbo Codes* est équivalent à la la propagation de croyances dans un modèle graphique contenant des cycles [KFL01].

3.4.3 ICM et Recuit simulé

Le cas le plus général du problème de minimisation implique

— une structure de graphe arbitraire ;
— des fonctions de potentiel d'énergie ayant une forme arbitraire ;
— des tailles arbitraires de l'ensemble $\mathcal{L}$ de valeurs possibles pour les variables.

Il est connu que ce problème est NP-difficile [KZ04]. Plusieurs possibilités existent d'obtenir une solution approchée. Une stratégie classique, appelée *Iterative Conditional Modes (ICM)*, procède de manière itérative en minimisant l'énergie pour chaque variable x_i séparément en considérant le reste du champ comme étant constant :

$$\widehat{x_i} = \arg\min_{x_i} E(x_i; x_{j\neq i}) \tag{3.28}$$

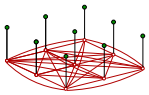
Le calcul de la minimisation peut être simplifié en incluant seulement les termes dépendant de la variable X_i. Cet algorithme glouton a le mérité d'être simple. Par contre, souvent les solutions obtenues sont de mauvaise qualité car elle dépendent fortement de l'initialisation du champ. De manière générale, seulement un minimum local est obtenu sans garantie de sa qualité.

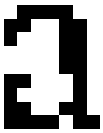
L'algorithme *Recuit Simulé* tente de résoudre ce problème en ajoutant une phase initiale d'exploration de l'espace de recherche. Le principe est similaire à l'algorithme ICM : le graphe est parcouru itérativement et chaque variable est mise à jour en fonction de la valeur actuelle de son voisinage. Par contre, là ou l'algorithme ICM procède de manière gloutonne, le recuit simulé, parfois, accepte des modifications entraînant une croissance de la fonction à minimiser. Des pas dans une « mauvaise direction » sont acceptés avec une probabilité q, qui dépend de la différence de l'énergie E avant et après la modification, et d'un paramètre T :

$$q = \exp\left\{-[E(x) - E(x')]/T\right\} \tag{3.29}$$

La « température » T va contrôler le degré de déterminisme du système [3]. Au début du processus d'optimisation cette valeur sera suffisamment grande pour fréquemment admettre des modifications des variables même si cela fait augmenter l'énergie. Au cours de l'optimisation la valeur de T sera diminuée, rendant l'algorithme de plus en plus glouton.

La qualité des solutions obtenues par le recuit simulé est généralement beaucoup plus élevée que celles des solutions obtenues par l'algorithme glouton ICM. Cet avantage est contre-balancé par une complexité de calcul accrue. Pour obtenir des très bonnes solutions, la vitesse de refroidissement du facteur de température T doit être suffisamment lente. Un des avantages significatifs de cet algorithme est son applicabilité générale. Aucune condition n'est imposée à la structure du graphe, à la forme de la fonction d'énergie, ou encore à la taille du domaine des variables.





Nous nous sommes servis du recuit simulé pour minimiser la fonction d'énergie d'un MRF avec un voisinage de très grande taille. La fonction d'énergie, décomposée en cliques de taille 16 (4×4) permet de restaurer les caractères d'une image de document [WD02]. Ces travaux moins récents n'ont pas été inclus dans ce mémoire.

3. Le nom « température » vient de la physique statistique, où ce genre d'algorithme est utilisé pour simuler le refroidissement de certains matériaux.

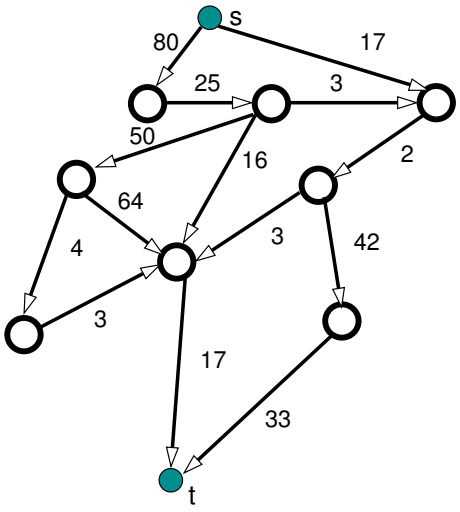


Figure 3.10 – Un exemple de graphe $s{-}t$ avec sa source s et son puits t.

3.4.4 Graph cuts / coupure minimale dans un graphe

La famille d'algorithmes dite *graph cuts* se sert de la découverte assez récente [DPS89, BVZ01, KZ04] que le minimum exact et global de certaines fonctions d'énergie peut être obtenu en calculant la coupure minimale dans un graphe de type $s{-}t$ (introduit dans la section suivante) construit spécifiquement pour le problème en question. Le problème général de minimisation étant NP-difficile [KZ04], cette solution est applicable à certains cas spécifiques seulement. Il ne s'agit pas d'un algorithme unique. Plusieurs techniques existent, chacune fournissant une solution pour une classe spécifique de fonctions. Selon la méthode choisie, des restrictions sont imposées sur la forme des potentiels d'énergie, et éventuellement aussi sur le nombre de valeurs possibles pour chaque variable, c.à.d. la taille du domaine. Pour certaines variantes le cas binaire est admis seulement. Dans le suite de cette section, nous présenterons d'abord le concept de la coupure minimale dans un graphe, et ensuite nous détaillerons quelques algorithmes de la famille *graph cuts*.

La coupure minimale dans un graphe de type $s{-}t$

Un graphe $s{-}t$ est un graphe orienté dont les arêtes $\mathcal{E}$ sont attribuées, et dont l'ensemble $\mathcal{V}$ des sommets inclut deux sommets particuliers appelés la source s et le puits t. Ce type de graphe est particulièrement adapté à la modélisation de problèmes impliquant un flux de la source s vers le puits t. Dans ce contexte, l'attribut associé à l'arête entre les sommets i et j modélise la capacité de l'arête à transporter un bien de i vers j, notée $c(i, j)$. La figure 3.10 donne un exemple d'un tel graphe.

Un concept important est celui de la coupure. La coupure d'un graphe $s{-}t$ est un ensemble C d'arêtes $C{\in}\mathcal{E}$ tel que la suppression des arêtes de la coupure du graphe initiale partitionne le graphe en deux ensembles non-connectés, et que la source s et le puits t se retrouvent dans deux ensembles différents. En d'autres termes, la coupure du graphe rend impossible de trouver un chemin entre s et t. Chaque sommet i dans le graphe peut alors être connecté à la source, écrit $i \in S$, ou il peut être connecté au puits, écrit $j \in T$.

La coupure minimale d'un graphe est une coupure tel que la somme des capacités de la coupure est minimale :

$$C^* = \arg\min_{C} \sum_{(i,j)\in C} c(i,j) \quad \text{tel que } C \text{ est une coupure du graphe } \mathcal{G} \tag{3.30}$$

Selon le théorème de Ford-Fulkerson, la valeur de la coupure minimale est égal au flux maximal du graphe, c.à.d. à la quantité de biens transportable de s vers t telle que les capacités c ne soient pas dépassées [FF62]. Les algorithmes de la famille *graph cuts* se servent du résultat selon lequel le calcul du flux maximal dans un graphe $s-t$ peut être effectué en un temps polynomiale. La complexité des meilleurs algorithmes est de $O(|\mathcal{V}|.|\mathcal{E}|^2)$ dans le pire des cas. Par contre, pour les graphes spécifiques issus des problèmes en vision par ordinateur, la complexité s'approche souvent du linéaire en fonction du nombre d'arêtes dans le graphe $s-t$ [BK04]. Rappelons ici que le graphe $s-t$ pour un problème donné ne correspond pas au graphe de dépendances. Selon le problème, le graphe $s-t$ peut avoir beaucoup plus de sommets et/ou d'arêtes que les graphes de dépendances.

Fonctions sous-modulaires et binaires

Une classe de fonctions d'énergie relativement commode est celle des fonctions *sous-modulaires*, binaires et dont la taille des cliques est restreint à 3. Ici, une fonction binaire signifie que chaque variable peut prendre deux valeurs possible, c.à.d. $\mathcal{L} = \{0, 1\}$. Le terme *sous-modulaire* sera expliqué plus tard dans cette section. Cette classe de fonctions a été étudiée d'abord par Boykov et Veksler [BVZ01] et ensuite par Kolmogorov et Zabih [KZ04]. Les derniers ont introduit une méthode pour la construction du graphe $s-t$ efficace qui est peut-être la plus utilisée à ce jour.

Ici nous présenterons deux cas simples pour illustrer le fonctionnement. Supposons d'abord un cas trivial impliquant deux variables : une variable cachée X et une variable observée Y liées par une fonction d'énergie simple composée d'un seul terme seulement : $E(x, y)$. Le graphe de dépendances est illustré dans la figure 3.11a. La minimisation est simple même sans algorithme complexe d'optimisation : l'espace des réalisations consiste en deux possibilités, $X = 0$ et $X = 1$, il suffit de choisir l'étiquette telle que l'énergie associée soit inférieure.

La minimisation par graph cuts consiste à construire le graphe $s-t$ montré dans la figure 3.11b. La solution est interprétée de la manière suivante : la variable correspond au seul sommet normal (différent de s et de t) du graphe. Si, après le calcul de la coupure, ce sommet reste connecté avec la source, alors $X = 1$, sinon $X = 0$. Pour que la coupure minimale corresponde à la solution du problème, il suffit donc de configurer les capacités des deux arêtes comme il est montré dans la figure 3.11b.

Généralisons maintenant le problème au cas de deux variables cachées X_1 et X_2 inter-dépendantes, avec un graphe de dépendances illustré dans la figure 3.12a. Comme ici nous sommes concernés par l'inférence des variables cachées, nous pouvons omettre de la notation (et du problème) les dépendances vers d'éventuelles variables observées sans perdre en généralité. Nous noterons donc la fonction d'énergie sur les deux variables comme $E(x_1, x_1)$. Dans le cadre le plus général (mais homogène), elle consistera de deux termes unaires (impliquant une seule variable) que nous noterons $E_1(.)$ et un terme par paire (impliquant deux variables) que nous noterons $E_2(.,.)$:

$$E(x_1, x_2) = E_1(x_1) + E_1(x_2) + E_2(x_1, x_2) \tag{3.31}$$

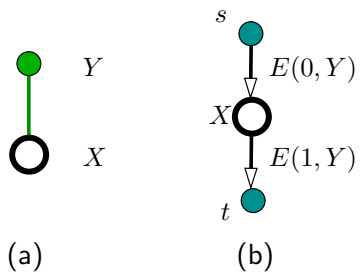


Figure 3.11 – (a) Un graphe de dépendances contenant une variable cachée et une variable observée ; (b) Le graphe $s{-}t$ correspondant à la minimisation de l'énergie de ce graphe (supposant un terme unique $E(x, y)$ dans l'énergie).

Pour des raisons pédagogiques nous évoquons d'abord le cas très classique où le fonctionnel $E_2(.,.)$ pénalise la différence d'étiquettes par un paramètre β et où il s'annule en cas d'égalité :

$$E_2(a, b) = \begin{cases} 0 & \text{if } a = b \\ \beta & \text{else} \end{cases} \tag{3.32}$$

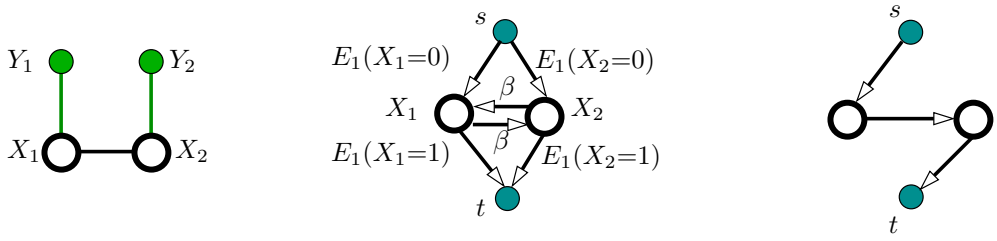
Ce modèle est aussi connu comme le modèle de Potts [Pot52, Li01], voir également la section 3.2.2 pour une application en segmentation d'images. L'énergie peut alors être minimisée par la méthode de construction de graphe de Boykov et Veksler [BVZ01]. Le graphe $s{-}t$ obtenu est donné dans la figure 3.12b. Chaque variable cachée correspond à un sommet dans le graphe, et chaque variable est reliée à la fois à la source s et au puits t, les capacités correspondant aux termes unaires. Les deux variables sont connectées entre elles par deux arêtes dont les capacités sont égales à β. Il est simple de voir que la coupure minimale correspond au minimum de l'énergie. Pour que l'algorithme puisse choisir une inégalité d'étiquettes, donc soit $X_1 = 0$ et $X_2 = 1$ ou l'inverse, une des deux arêtes de capacité β doit être coupée, sinon un flux est possible entre s et t.

Généralisons maintenant le comportement à d'autres formes pour le fonctionnel $E_2(.,.)$. Dans le cadre plus général, l'énergie E, pourtant simple, peut-être minimisée de manière exacte par une coupure minimale, si le fonctionnel choisi pour le terme $E_2(.,.)$ est sous-modulaire, c.à.d. s'il satisfait la propriété suivante :

$$E_2(0, 0) + E_2(1, 1) \leq E_2(0, 1) + E_2(1, 0) \tag{3.33}$$

En d'autres termes, la fonction d'énergie est sous-modulaire si l'égalité d'étiquettes est favorisée par rapport à l'inégalité — sans qu'il soit nécessaire qu'elle s'annule en cas d'égalité, comme cela avait été demandé pour (3.32). Notons ici que ce genre de contraintes est naturellement satisfait pour un grand nombre d'applications, notamment la segmentation d'images, où l'objectif consiste à favoriser l'égalité d'étiquettes pour des pixels voisins. En revanche, si les frontières entre les régions sont modélisées, l'égalité des étiquettes doit être favorisée dans certains cas uniquement, ce qui peut rendre certains termes non sous-modulaires. Un tel modèle est présenté dans le chapitre 6, section 6.2.2 sur la décomposition de maillages 2-variété.

Une méthode a été introduite par Kolmogorov et Zabih [KZ04] pour construire des graphes pour cette classe de fonctions. Elle est généralement plus efficace que la méthode de Boykov et Veksler [BVZ01], aboutissant à des graphes ayant moins d'arêtes. Le graphe $s{-}t$ construit pour l'exemple ci-dessus, c.à.d. pour l'énergie donnée dans (3.31), est donné dans la figure 3.12c. Pour



(a) (b) (c)

Figure 3.12 – (a) un graphe de dépendances contenant deux variable cachées et deux variables observées ; (b) le graphe $s{-}t$ correspondant à la minimisation de l'énergie de ce graphe, construit avec la méthode de Boykov et Veksler [BVZ01] ; (c) le graphe construit avec la méthode de Kolmogorov et Zabih [KZ04].

plus de détails sur le calcul des capacités des arêtes du graphe $s-t$ nous renvoyons le lecteur à [KZ04]. Notons seulement que la structure du graphe, donc l'existence d'arêtes entre une variable et la source ou le puits, respectivement, dépend des valeurs des potentiels unaires.

La méthode permet également de construire des graphes pour des fonctions d'énergie impliquant des termes ternaires, c.à.d. des termes sur des triplets de variables. Elle demande alors l'introduction d'un sommet auxiliaire dans le graphe $s{-}t$ par terme ternaire.

Fonctions sous-modulaires avec plusieurs labels : mouvement d'expansion-α

Si pour un modèle le nombre d'étiquettes admises pour les variables dépasse les deux, les deux méthodes de construction de graphe $s{-}t$ décrites ci-dessus ne permettent pas de minimiser la fonction d'énergie. L'algorithme *mouvement d'expansion-α* est une méthode itérative et approchée traitant les problèmes impliquant un nombre arbitraire d'étiquettes [BVZ01, KZ04]. L'idée est la suivante : après une initialisation, un sous-problème est traité dans chaque itération. Deux possibilités sont alors proposées pour chaque sommet, donc pour chaque variable :

— garder la valeur actuelle ;
— basculer la valeur à α,

où α est une valeur constante durant une itération donnée. La valeur de α est changée à chaque itération. Pour chaque sous-problème, un graphe $s{-}t$ est construit permettant de trouver la solution optimale.

Comme pour l'algorithme ICM (voir la section 3.4.3), l'algorithme converge vers un optimum local uniquement. Par contre, de manière générale cet optimum est d'une bien meilleure qualité. Cela s'explique par la taille du mouvement optimal calculé à chaque itération. En effet, les deux algorithmes, ICM et mouvement d'expansion-α, procèdent par un algorithme itératif calculant des séquences de mouvements optimaux, sans pour autant pouvoir garantir l'optimalité du résultat global. Par contre, là ou ICM propose des mouvements impliquant un sommet à la fois, l'optimalité se résumant au calcul d'une étiquette par rapport aux valeurs des voisins, le mouvement du deuxième algorithme permettra de potentiellement changer le graphe entier de manière optimale — restreint au basculement vers l'étiquette α proposé, bien évidemment.

Notons que, pour une itération donnée, l'étiquette α doit être la même pour tous les sommets du graphe. Cela est nécessaire pour que le sous-problème reste sous-modulaire. Cette faiblesse sera

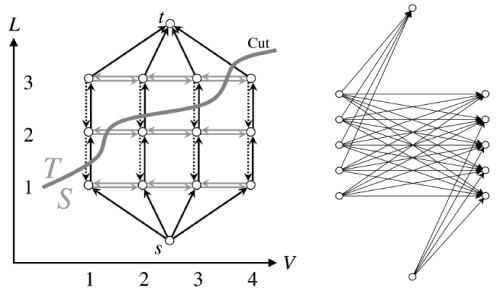


Figure 3.13 – Construction d'un graphe $s{-}t$ pour la minimisation exacte de fonctions dont la différence d'étiquettes est convexe. Illustration reproduite de [Ish03].

adressée par le mouvement de mélange, un algorithme présenté plus tard dans cette section.

Fonctions de différences d'étiquettes convexes (multiples)

Une classe de fonctions souvent rencontrées lors de problèmes de restauration d'images concerne les fonctions multi-étiquettes où le terme par paire s'exprime comme une fonction convexe définie sur la différence d'étiquettes :

$$E_2(x_i, x_j) = g(x_i - x_j) \tag{3.34}$$

où $g(.)$ est une fonction convexe. La méthode de construction de graphe proposée par Ishikawa [Ish03] permet de trouver une solution globale exacte sans passer par un algorithme itératif de type mouvement d'expansion-α. Le graphe construit contient plusieurs sommets par sommet du graphe d'origine, comme illustré dans la figure 3.13 : pour chaque étiquette admise par un sommet du graphe d'origine, un sommet est ajouté au graphe $s{-}t$, les différentes étiquettes étant liées sous forme d'une chaîne connectée à la source s et au puits t par les extrémités. Il est facile de voir qu'une coupure du graphe coupera chaque chaine au moins une fois, exactement une fois pour la coupure minimale. Si les capacités des arêtes sont judicieusement choisies, la coupure de ce graphe correspond au minimum de la fonction d'énergie. Le résultat est interprété de la manière suivante : la variable donnée prendra l'étiquette qui correspond au sommet au dessus de la coupure dans le graphe.

Fonctions non sous-modulaires et binaires

Il est connu que la minimisation de fonctions d'énergie arbitraires, donc en particulier non sous-modulaires, est NP-difficile [KZ04] :

Théorème 2 *Soit $E_2(.,.)$ une fonction sur deux variables. La minimisation de fonctions de forme*

$$E(x_1, \ldots, x_N) = \sum_i E_1(x_i) + \sum_{(i,j)\in\mathcal{E}} E_2(x_i, x_j) \tag{3.35}$$

où les E_i sont des fonctions arbitraires et où $\mathcal{E} \in N \times N$, est NP-difficile.

Nous renvoyons le lecteur à [KZ04] pour la preuve de ce théorème.

La minimisation approchée de ce genre de problème a été abordée dans la communauté de mathématiques discrètes depuis 30 à 40 ans, où le problème est connu sous le nom de *Quadratic Pseudo-Boolean Optimisation* ou QPBO [HHS84]. Les communautés de vision par ordinateur et de la modélisation par graphes ont récemment ré-decouvert ce problème et les approches connues. Ici nous donnerons les grandes lignes d'un algorithme maintenant appelé QPBO et publié par Kolmogorov et Rother dans un journal de vision par ordinateur [KR07]. D'autres variantes ont été découvertes depuis [RKLS07].

Le graphe $s{-}t$ construit pour un problème de ce type contient deux sommets par variable : un sommet correspond à la valeur même de la variable, c.à.d. à X_i, et un sommet correspond à son complément $\overline{X_i}$. La figure 3.14a montre un exemple d'un graphe de dépendances impliquant deux variables cachées. Comme auparavant, nous pouvons ignorer les dépendances vers les variables observées pour l'étape d'inférence des variables cachées. La figure 3.14b montre un exemple d'un graphe $s{-}t$ construit avec la méthode QPBO. La structure exacte du graphe $s{-}t$ dépend de l'énergie associée au graphe de dépendances, tout particulièrement des potentiels d'interactions entre les deux variables.

Comme chaque variable X_i est présentée deux fois, une fois de manière directe et une fois par son complément $\overline{X_i}$, 4 résultats sont possibles pour chaque variable puisque chacun des deux sommets X_i ou $\overline{X_i}$ peut être connecté avec la source s ou avec le puits t. Si l'énergie est sous-modulaire, les résultats sont cohérents, c.à.d. si $X_i \in S$ alors $\overline{X_i} \in T$, ce qui veut dire que $X_i = 0$. Dans le cas inverse $X_i = 1$.

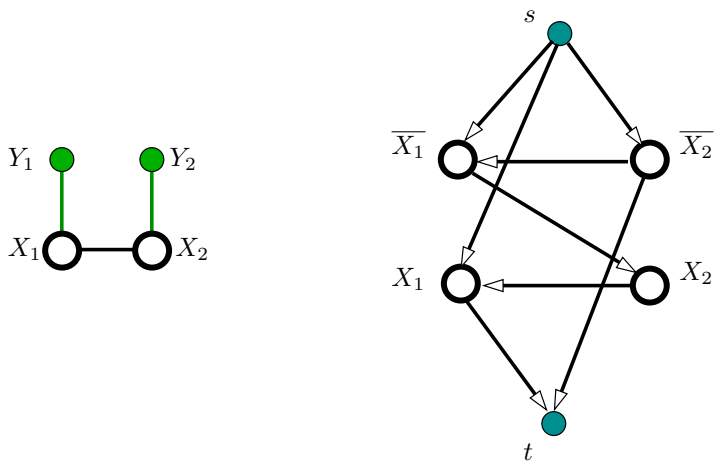
Si l'énergie est non sous-modulaire (rappelons nous que le problème est NP-difficile dans ce cas), il peut y avoir une incohérence entre les deux sommets. QPBO donne alors un étiquetage partiel, c.à.d. pour certaines variables le résultat n'est pas connu. Le nombre de variables non étiquetées n'est pas connu d'avance et ne peut pas être dérivé facilement à partir du nombre de termes non sous-modulaires de l'énergie. Or, dans les applications que nous avons traitées dans ce mémoire, très peu de variables non étiquetées ont été observées expérimentalement — voir le chapitre 6.

L'intérêt de la méthode QPBO réside dans deux propriétés importantes [KR07] :

Théorème 3 (Optimalité partielle) *Soit x' une solution obtenue par QBPO pour une fonction de forme (3.35). Il existe un minimum global x^* tel que $x'_i = x^*_i$ pour toutes les variables x'_i étiquetées, c.à.d. pour toutes les $x'_i \in \{0, 1\}$.*

Théorème 4 (Persistance) *Soit x' une solution obtenue par QBPO pour une fonction de forme (3.35) ; soit x^c un étiquetage complet ; Soit x^f la fusion des deux étiquetages, tel que $x^f_i = x'_i$ si $x'_i \in \{0, 1\}$ et $x^f_i = x^c_i$ sinon. Alors $E(x^f) \leq E(x^c)$.*

La première propriété garantit que les étiquettes trouvées par la méthode sont les étiquettes optimales indépendamment du fait que l'étiquetage obtenu soit complet ou partiel. La deuxième propriété garantit que l'amélioration d'un étiquetage existant par un étiquetage obtenu par QPBO ne diminuera pas la qualité de la solution. Ensemble, cela permet de fusionner facilement des résultats obtenus avec plusieurs modèles ou méthodes avec une garantie de convergence — voir le paragraphe suivant.



(a) (b)

Figure 3.14 – (a) Un exemple de graphe de dépendances sur deux variables cachées ; (b) son graphe $s{-}t$ construit avec la méthode QPBO. La structure exacte du graphe $s{-}t$ dépend de l'énergie associé au graphe de dépendances.

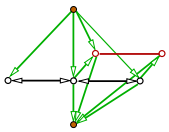
Fonctions non sous-modulaires avec plusieurs labels : mouvement de mélange

Comme nous l'avions mentionné, l'algorithme *mouvement d'expansion-α* décrit ci-dessus propose une nouvelle étiquette unique α à toutes les variables à chaque itération. Cette proposition unique permet de garantir la sous-modularité du sous-problème.

L'algorithme *mouvement de mélange* (*Fusion Move*) permet de minimiser, de manière approchée, les fonctions d'énergie à plusieurs étiquettes, qu'elles soient sous-modulaires ou non sous-modulaires [LRRB10]. L'idée est similaire à l'algorithme d'expansion-α : le champ de variables est initialisé et un algorithme itératif propose des nouvelles étiquettes aux variables à chaque itération. Par contre, la proposition d'étiquette est adaptée à chaque variable, ce qui rend le sous-problème de chaque itération non sous-modulaire. Une solution est trouvée avec l'algorithme QPBO, qui donne une solution partielle incluant des variables non étiquetées (voir le paragraphe précédent). Les propriétés de persistance (Théorème 4) et d'optimalité partielle (Théorème 3) permettent de garantir la convergence de l'algorithme.

Deux questions restent ouvertes : que fait-on avec les variables non étiquetées et comment les propositions individuelles d'étiquettes sont-elles calculées ? Les premières sont habituellement gardées inchangées entre les itérations. En ce qui concerne les propositions, plusieurs possibilités ont été essayées : des recherches locales, des propositions par d'autres algorithmes d'optimisation discrète comme la propagation de croyances, la décomposition numérique des étiquettes bit par bit (*Log Cut*, [LRRB10]) etc. L'algorithme a également été appliqué à l'optimisation continue-discrète en proposant une séquence de candidats pour des problèmes continues [LRR08] — voir aussi nos

travaux sur l'optimisation de maillages 3D surfaciques 2-variété dans le chapitre 6.



Dans ce mémoire nous nous servons de l'optimisation par graph cuts dans de nombreux travaux : segmentation d'images de documents, segmentation de vidéo et estimation de modèles de mouvement, segmentation de maillages, optimisation et simplification de maillages 2-variété etc.

Inférence exacte par propagation dans les arbres de jonction

De manière générale, l'inférence est difficile pour les modèles graphiques combinant deux propriétés :

— la structure est générale, c.à.d. le graphe de facteurs associé à la fonction d'énergie contient des cycles ;

— les fonctionnelles de cliques sont de formes arbitraires.

Cependant, même pour un tel modèle, l'inférence peut avoir une faible complexité de calcul. L'algorithme par propagation dans les arbres de jonction (*junction tree algorithm*) permet l'inférence exacte dans les modèles graphiques à structure générale avec une complexité exponentielle dans le pire des cas [LS88]. Or, la complexité peut être faible, si après triangulation du graphe [4], la taille de la plus grande clique reste faible. Nous ne détaillerons pas cet algorithme dans ce mémoire.

Modèles graphiques, optimisation discrète et topologie

Les modèles graphiques permettent de modéliser des problèmes en obtenant des solutions cohérentes. Pour cela, des connaissances locales et des connaissances sur les interactions entre les différentes variables sont injectées. Par contre, dans certaines applications on souhaiterait obtenir des solutions satisfaisant certaines propriétés topologiques. Typiquement, et pour le cas d'images 2D, des contraintes sur le nombre de composantes connexes et sur le nombre de trous pourraient être données par un utilisateur. Des contraintes similaires s'appliquent au cas d'images n-D. Ce problème revient à une minimisation sous contraintes :

$$\begin{aligned} \widehat{x} = \quad & \arg\min_{x} \sum_{i} E_1(x_i) + \sum_{(i,j)\in\mathcal{E}} E_2(x_i, x_j) \\ & \text{tel que } \mathcal{T}(x) = \mathcal{T}_d \end{aligned} \tag{3.36}$$

où $\mathcal{T}(x)$ dénote la topologie du champ x et $\mathcal{T}_d$ est la topologie demandée par l'utilisateur. Ce problème est NP-complet même pour des fonctions d'énergie sous-modulaires, et même pour des fonctions d'énergie comportant des potentiels unaires seulement [ZSCP08, CFL11].

A notre connaissance, seulement deux méthodes approchées ont été proposées pour ce problème. Les *Topology cuts* consistent en une modification directe de l'algorithme Ford-Fulkerson pour le calcul de la coupure minimale dans un graphe $s{-}t$ [ZSCP08]. Une étiquette pour chaque

4. Trianguler un graphe implique l'ajout d'arêtes tant qu'il existe au moins un cycle de longueur > 3 sans corde, c.à.d. sans arête reliant deux sommets non-adjacents du cycle.

sommet du graphe $s{-}t$ modélise de manière explicite l'appartenance du sommet à S ou T. Les contraintes topologiques sont gardées intacte durant le calcul du flux, ce qui est possible seulement en calculant une approximation du flux maximal.

Dans [CFL11], une autre stratégie est poursuivie. Un algorithme itératif résout le problème à l'aide d'un algorithme de *graph cuts* classique en ignorant les contraintes topologiques. Ensuite, la fonction d'énergie est modifiée en adaptant les potentiels unaires pour obtenir une nouvelle solution satisfaisant plus de contraintes topologiques que la solution précédente. Typiquement, le nombre de composantes connexes et le nombre de trous sont plus proches aux nombres souhaités. La méthode cherche à minimiser les modifications apportées aux termes unaires afin de changer les propriétés de la solution le moins possible. Malheureusement la recherche de ce minimum est également NP-complet pour la plupart des métriques mesurant la quantité de modifications. Une solution peut être trouvée pour la métrique L^∞ en temps polynomial. A titre d'exemple nous citons le problème consistant à fusionner deux composantes connexes. En cherchant le plus court chemin entre les deux composantes et en perturbant les termes unaires sur ce chemin, l'algorithme itératif convergera vers une solution fusionnant les deux composantes connexes.

Les deux méthodes ne produisant pas des solution optimale. La méthode [ZSCP08] donne des minima locaux pas toujours de bonne qualité. La méthode [CFL11] produit des images localement très perturbées par les modifications des termes unaires. L'erreur produite n'est pas distribuée de manière équilibrée.

3.5 Inférence des paramètres : apprentissage

Les fonctionnelles d'un modèle graphique (potentiels de cliques pour les MRF ou probabilités conditionnelles pour les BN) sont habituellement définies en tant que fonctions paramétriques [5]. Avant de pouvoir estimer les variables cachées, il faut donc estimer les paramètres de ces fonctionnelles. Les quatre sous-sections suivantes donnent une (très) courte introduction à l'apprentissage de paramètres. La difficulté du problème dépend

— du type de modèle — génératif ou discriminatif ;
— de la disponibilité de réalisations pour les variables cachées dans les données d'apprentissage, ce qui se traduit en un problème supervisé ou non-supervisé ou éventuellement faiblement supervisé si une partie des données est disponible ;
— de la structure du graphe — linéaire, sous forme d'arbre, ou général, c.à.d. contenant des cycles.

Comme pour l'inférence des étiquettes, une structure générale du graphe n'implique pas forcement une complexité élevée — voir la section 3.4.4. D'ailleurs, pour une partie des algorithmes d'apprentissage, les algorithmes d'inférence des étiquettes cachées décrites ci-dessus sont appelés pour résoudre un sous-problème intervenant dans l'apprentissage des paramètres.

3.5.1 Estimation supervisée — modèles génératifs

Dans le cas de l'estimation supervisée, les données d'apprentissage comprennent à la fois des réalisations des variables observées y et des variables cachées x. La façon la plus naturelle pour

5. Ici nous ne traiterons pas le cas où les fonctionnelles sont discrètes et tablées et où les entrées des tables sont apprises à partir des données d'apprentissage [MS93, WD02].

apprendre les paramètres d'un modèle probabiliste, qu'il soit graphique ou pas, est de maximiser la vraisemblance des données d'apprentissage. Pour les modèles génératifs, il s'agit de la vraisemblance suivante, basée sur la probabilité jointe :

$$\mathcal{L}_{G^*}(\theta) = \ln p(x, y|\theta) \tag{3.37}$$

où x et y sont, respectivement, les données d'apprentissage pour les étiquettes cachées et pour les observations. Grâce aux indépendances conditionnelles (3.18) habituellement supposées pour les modèles génératifs, les paramètres θ_1 du *prior* $p(x)$ et les paramètres θ_2 du modèle d'observation $p(y|x)$ peuvent être appris de manière séparée. Ici nous ne traiterons pas le cas des paramètres du modèles d'observation, plutôt faciles à estimer sous contraintes (3.18). Les paramètres du *prior* dépendent uniquement de la réalisation du champ caché, ce qui donne la vraisemblance suivante :

$$\mathcal{L}_G(\theta_1) = \ln p(x|\theta_1) \tag{3.38}$$

Pour les MRF, la difficulté majeure est le calcul de la fonction de partition $Z(\theta_1)$, qui dépend des paramètres à apprendre θ_1 :

$$Z(\theta_1) = \sum_x \exp\{-E(x;\theta_1)\} \tag{3.39}$$

où $E(x)$ est la fonction d'énergie restreint aux cliques impliquant les variables cachées. La maximisation de (3.38) est difficile, car le calcul de $Z(\theta_1)$ est impossible même pour des problèmes assez simples. La plupart des approximations sont basées sur la probabilité conditionnelle suivante impliquant un site i et son voisinage N_i :

$$p(x_i|x_{N_i}, \theta_1) = \frac{\exp\{-E(x_i, x_{N_i};\theta_1)\}}{\sum_{x'_i} \exp\{-E(x'_i, x_{N_i};\theta_1)\}} \tag{3.40}$$

Ici $E(x_i, x_{N_i};\theta_1)$ sont les termes de l'énergie $E(x)$ comprenant le site i. Notons que cette probabilité ne dépend pas de $Z(\theta_1)$.

La *pseudo-likelihood* [Bes75] approxime la vraisemblance (3.38) par un produit de (3.40) sur tous les sites i :

$$\mathcal{PL}_G(\theta_1) = \ln \prod_i p(x_i|x_{N_i}, \theta_1) \tag{3.41}$$

La méthode de coding partitionne un MRF en partitions statistiquement indépendantes et résout (3.41) de manière indépendante pour chaque partition [Bes74]. L'approximation de type *meanfield* remplace une réalisation x_{N_i} dans (3.41) par les moyennes $< X_{N_i} >$ des variables aléatoires X_{N_i}, pour lesquelles une approximation est également nécessaire [Par88]. La méthode par moindres carrés [DEK85] établie une relation linéaire entre les paramètres θ_1 et la fonction $E(x_i, x_{N_i}, \theta_1)$ de (3.40) :

$$E(x_i, x_{N_i}, \theta_1) = \theta_i^T F(x_i, x_{N_i}) \tag{3.42}$$

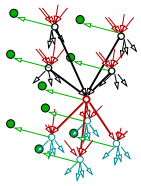
En utilisant (3.40), pour une paire de sites x_i et x'_i ayant la même réalisation du voisinage x_{N_i}, la relation suivante peut être établie :

$$\theta^T[F(x'_i, x_{N_i}) - F(x_i, x_{N_i})] = \ln\left(\frac{p(x_i|x_{N_i})}{p(x'_i|x_{N_i})}\right) \tag{3.43}$$

En remarquant que la probabilité $p(x_{N_i})$ est la même pour les deux sites x_i et x'_i, nous pouvons voir que les probabilités conditionnelles sont égales aux probabilités absolues :

$$\theta^T[F(x'_i, x_{N_i}) - F(x_i, x_{N_i})] = \ln\left(\frac{p(x_i, x_{N_i})}{p(x'_i, x_{N_i})}\right) \tag{3.44}$$

La partie droite de (3.44) peut être estimée à partir des données d'apprentissage avec des techniques basées sur les histogrammes. En cherchant cette relation pour des étiquetages différents, on obtient un système sur-déterminé d'équations linéaires qui peut être résolu par des méthodes de moindres carrés.



Dans ce mémoire nous nous servons de cette méthode pour estimer les paramètres de notre MRF à double couche. Nous nous sommes également inspirés de cette méthode pour concevoir une approche similaire pour l'estimation des paramètres du notre modèle de cube de Markov, un réseau Bayesien — voir le chapitre 5.

L'estimation par la méthode MCMC (*Markov chain Monte Carlo*) passe par un échantillonnage itératif de la distribution $p(\theta_1|x) \propto p(\theta_1)p(x|\theta_1)$, où le *prior* sur les paramètres $p(\theta_1)$ peut être uniforme. Une chaîne de Markov est créée, qui passe d'un état θ_1 au prochain θ'_1 avec une probabilité de transition garantissant que l'équilibre correspond à la distribution ciblée. Dans [WLL00], la probabilité de transition est celle de Metropolis-Hastings,

$$\alpha(\theta_1, \theta') = \min\left(1, \frac{p(x|\theta'_1)}{p(x|\theta_1)}\right) \tag{3.45}$$

où $p(x|\theta_1)$ est approximée par la *pseudo likelihood* (3.41)

3.5.2 Estimation non-supervisée — modèles génératifs

Pour la majorité des applications, les données d'apprentissage comprennent des réalisations du champ observé Y mais pas du champ caché X. Une possibilité intuitivement logique est d'initialiser le champ X avec une méthode alternative simple, et d'alterner deux étapes : i) estimation des paramètres θ_1 à partir de la réalisation actuelle du champ X ; ii) estimation du champ X à partir de la valeur actuelle de θ_1.

Une telle stratégie est mis en oeuvre par l'algorithme *Iterated Conditional Estimation* (ICE) [BP93, Pie07], qui dépend de deux estimateurs supervisés $\Phi_1(x)$ et $\Phi_2(x, y)$ capables d'estimer les paramètres à partir de l'ensemble complet des données — voir la section 3.5.1. A chaque itération, les nouvelles valeurs $\theta^{[t+1]}$ pour les paramètres sont estimées comme l'espérance conditionnelle suivante basée sur la valeur actuelle $\theta^{[t]}$ des paramètres :

$$\begin{aligned} \theta_1^{[t+1]} &= E_{\theta^{[t]}}[\Phi_1(x)|y] \\ \theta_2^{[t+1]} &= E_{\theta^{[t]}}[\Phi_2(x, y)|y] \end{aligned} \tag{3.46}$$

En pratique, cette espérance est difficile à calculer. Une solution est de l'approximer par une moyenne d'échantillons de $p(x|y, \theta^{[t]})$ calculés par une méthode comme l'échantillonneur de Gibbs [MCPB00].

Un principe similaire est utilisé dans l'estimation de paramètres par l'algorithme EM (*Expectation-Maximization*), un algorithme itératif similaire initialement proposé dans un contexte plus général [DLR77]. L'étape « E » calcule les paramètres de l'espérance conditionnelle suivante :

$$\begin{aligned} Q(\theta|\theta^{[t]}) &= E_{d,\theta^{[t]}}[\ln p(x,y|\theta)] \\ &= E_{d,\theta^{[t]}}[\ln p(y|x,\theta) + \ln p(x|\theta)] \\ &= \sum_x \{\ln p(y|x,\theta) + \ln p(x|\theta)\}\, p(x|y,\theta^{[t]}) \end{aligned} \tag{3.47}$$

L'étape « M » maximise cette espérance pour obtenir une nouvelle estimation des paramètres θ :

$$\theta^{[t+1]} = \arg\max_\theta Q(\theta|\theta^{[t]}) \tag{3.48}$$

Le calcul est assez facile si le graphe de dépendances n'a pas de cycle, comme dans le cas des HMM, pour lesquels l'algorithme EM est équivalent à l'algorithme Baum-Welch [Rab89], ou pour les arbres quaternaires de Markov [LPH00], voir le chapitre 5. Dans le cas général, des approximations sont nécessaires, comme par exemple celles décrites dans la section 3.5.1 — e.g. la *pseudo-likelihood* ou le *mean field* [Zha92].

3.5.3 Estimation supervisée — modèles discriminatifs

Ici nous sommes concernés par les modèles discriminatifs de type CRF, pour lesquels l'estimation des paramètres $\theta{=}\{\lambda_{ck}\}$ (voir la section 3.3) par maximum de vraisemblance recourt à la vraisemblance conditionnelle :

$$\mathcal{L}_D(\theta) = \ln p(x|y,\theta) \tag{3.49}$$

Comme pour les MRF génératifs, la fonction de partition Z dépend des paramètres θ, mais contrairement aux MRF génératifs, elle dépend également des données y ; dans la suite nous la noterons donc $Z(y,\theta)$. L'apprentissage est effectué sur un ensemble de N échantillons indépendants $(x^{(i)}, y^{(i)})$, ce qui nous emmène à

$$\mathcal{L}_D(\theta) = \ln \prod_i p(x^{(i)}|y^{(i)},\theta) \tag{3.50}$$

En combinant la forme des CRF (3.19) et la vraisemblance ci-dessus (3.49) et en ajoutant des termes de régularisation, nous obtenons

$$\mathcal{L}_D(\theta) = \sum_{i=1}^{N} \sum_{c\in\mathcal{C},k} \lambda_{ck} f_k(x_c^{(i)}, y^{(i)}) - \sum_{i=1}^{N} \ln Z(y^{(i)},\theta) - \sum_{c,k} \frac{\lambda_{ck}^2}{2\sigma^2} \tag{3.51}$$

où les σ^2 sont les variations d'un *prior* Gaussien de moyenne zéro sur les paramètres λ_{ck}. Nous remarquons que la fonction de partition $Z(y^{(i)},\theta)$ est différente pour chaque échantillon $y^{(i)}$. Contrairement aux MRF génératifs, la minimisation doit être effectuée séparément pour chaque échantillon.

De plus, contrairement aux MRF plus généraux, dans le cas des CRF, les paramètres $\theta{=}\{\lambda_{ck}\}$ se présentent sous une forme bien spécifique, à savoir sous forme de poids pour les fonctions f_k associées aux cliques c. Les fonctions f_k ne peuvent comprendre aucun paramètre supplémentaire. Cette forme rend la vraisemblance (3.51) convexe sur les paramètres θ, ce qui permet l'application

d'algorithmes numériques classiques d'optimisation continue pour obtenir un maximum/minimum global. Ce calcul est facile pour les CRF à structure linéaire ou sous forme d'arbre, pour lesquels une solution globale pour (3.51) peut être trouvée. Pour les CRF à structure de graphe arbitraire, l'apprentissage est difficile de manière générale, nécessitant le recours aux méthodes approchées.

3.5.4 Estimation non-supervisée — modèles discriminatifs

Comme pour les modèles génératifs, dans le cas de l'estimation non-supervisée, les variables cachées X ne sont pas connues lors de l'apprentissage. Dans la pratique, souvent les étiquettes pour une partie des variables sont connues, tandis que les valeurs pour une partie complémentaire sont inconnues. A titre d'exemples nous pouvons citer le HCRF [QWM+07] introduit dans la section 3.3 pour la classification de séquences, et illustré dans la figure 3.9b. Lors de l'apprentissage, les étiquettes pour la variable U (la classe de la séquence) sont connues, alors que les étiquettes pour les états cachés X de la « chaîne » ne le sont pas.

De manière générale, nous pouvons séparer l'ensemble X en deux sous-ensembles : l'ensemble X pour lequel les étiquettes sont connues durant la phase d'apprentissage, et l'ensemble U, pour lequel les étiquettes ne sont pas connues. La vraisemblance à maximiser pour l'estimation des paramètres θ doit marginaliser sur les variables U, ce qui donne l'équation suivante (pour un seul échantillon et sans régularisation des paramètres θ) :

$$\mathcal{L}_D(\theta) = \ln p(x|y) = \ln \sum_u p(x,u|y) = \ln \left\{ \frac{1}{Z(y^{(i)},\theta)} \sum_u \prod_{c\in\mathcal{C},k} \lambda_{ck} f_k(x_c,u_c,y) \right\} \tag{3.52}$$

Contrairement à (3.51), la fonction (3.52) n'est pas convexe. Quelque soit le graphe, linéaire, sous forme d'arbre, ou avec une structure générale, les algorithmes d'optimisation peuvent donc trouver un maximum/minimum local uniquement, dont la qualité va dépendre de l'initialisation.

Pour les CRF à structure linéaire ou sous forme d'arbre, plusieurs méthodes de minimisation existent, parmi lesquelles nous citerons celle basée sur la descente de gradient. En effet, les dérivées de la vraisemblance (3.52) par rapport aux paramètres θ peuvent être exprimées (après quelques opérations algébriques) comme suit [QCD05, SM12] :

$$\begin{aligned} \frac{\mathcal{L}_D(\theta)}{\lambda_{ck}} &= \sum_{c\in\mathcal{C},k} \sum_{u'_c} p(u'_c|x,y) \lambda_{ck} f_k(x_c,u'_c,y) \\ &- \sum_{c\in\mathcal{C},k} \sum_{u'_c,x'_c} p(u'_c,x'_c|y) \lambda_{ck} f_k(x_c,u'_c,y) \end{aligned} \tag{3.53}$$

Les probabilités marginales $p(u'_c|x,y)$ et $p(u'_c,x'_c|y)$ sont faciles à calculer pour les CRF à structure linéaire ou sous forme d'arbre. Pour les CRF à structure générale, elles peuvent être approximées avec la version non-exacte de la propagation de croyances (*Loopy Belief Propagation*).

Certaines méthodes développées pour les modèles génératifs à structure générale, et présentées dans la section 3.5.1, sont applicables, telles que la *pseudo-likelihood* [Bes75]. Remarquons que la méthode des moindres carrés [DEK85] pour les modèles génératifs ne peut être appliquée. En effet, la dépendance potentielle de chaque fonction f_k de toutes les observations y du modèle rend difficile l'estimation des probabilités $p(x,y,\theta')$, remplaçant $p(x,\theta'_1)$ dans (3.44), par des méthodes d'histogrammes.

3.6 Comparaison de quelques familles de modèles connues

Dans ce chapitre nous avons essayé de donner une introduction à la modélisation par modèles graphiques probabilistes. Dans la figure 3.15 nous avons mis en relations quelques modèles connus en décrivant leur différences. De manière générale, les différences se déclinent selon

— le type de graphe, orienté (causal) ou non orienté (non-causal) ;
— la structure du graphe : chaîne, arbre, structure générale et cyclique, structure cyclique et adaptée à une application spécifique etc. ;
— le type de modèle, génératif ou discriminatif.

Les contributions décrites ont été développées durant plusieurs décennies et au sein de plusieurs communautés scientifiques, parfois en parallèle. Pour cette raison, les termes ne sont pas toujours bien définis. Ainsi, le terme *Modèle de Markov Caché (HMM)* devrait logiquement s'appliquer à tout modèle dont le champ caché est Markovien. Or, pour des raisons historiques, seulement les modèles génératifs ayant une structure linéaire (une chaîne) portent ce nom.

Une version discriminative des HMM basée sur un graphe non-orienté a été introduit il y a une dizaine d'années [LMP01]. Cette famille de modèle, nommée *Champs de Markov Conditionnelles (Conditional Random fields, CRF)* était donc à l'origine conçue pour des structures linéaires. Or, rapidement le terme a été employé pour des extensions aux graphes de structure arbitraire. Un CRF peut donc être vu de plusieurs manières :

— comme une version discriminative d'un HMM sur graphe non-orienté avec éventuellement une généralisation de la structure à des graphes arbitraires ;
— comme une version discriminative d'un MRF (qui, lui, est déjà défini sur graphes non-orientés).

Un HMM peut être considéré comme une version spécifique d'un réseau Bayesien où le graphe de dépendances est déroulé dans le temps et qui donc suit une forme linéaire, c.à.d. une chaîne. Malgré cette similarité, l'histoire des deux modèles est différente, puisque les HMM ont été surtout développés dans la communauté de traitement de signal, alors que les réseaux Bayesiens sont un outil surtout utilisé par la communauté informatique et intelligence artificielle. Il existe d'ailleurs des modèles intermédiaires portant deux noms différents, à savoir *HMM couplés* ou *Réseaux Bayesiens Dynamiques* [BOP97]. Il s'agit du même modèle, appliqué dans des situations où l'on souhaite gérer de multiples séquences de mesures avec des interdépendances. A titre d'exemple nous pouvons citer le suivi de personnes dans un environnement multi-caméra. La structure graphique du modèle n'est pas linéaire simple comme pour les HMM. Par contre, elle n'est pas non plus dépourvue de toute régularité : il s'agit de deux (ou plusieurs) chaînes connectées tel qu'un sommet correspondant à un instant temporel t est connecté à tous les sommets correspondant aux instant $t-1$ et $t+1$.

Une autre extension des HMM permet de gérer des séquences d'observations ayant un comportement plus complexe. Les contraintes classiques de Markovianité de la chaîne cachée et d'indépendance conditionnelle des observations sachant les états (3.18) sont assez fortes ; un grand nombre de phénomènes physiques ne peut être modélisé sous de telles contraintes, comme par exemple des mélanges de textures etc. Les chaînes de Markov couples [Pie03] (*Pairwise Markov Chaines*, à ne pas confondre avec les *Coupled Markov Chains* décrites ci-dessus) traitent ce problème en relâchant les contraintes de Markovianité de la chaîne cachée. Elles sont remplacées par des contraintes de Markovianité de la chaine couple cachée-observée. Ce concept peut être encore étendu à des chaînes triplets, permettant d'inclure une chaîne de paramètres dans la séquence [LLLP08].

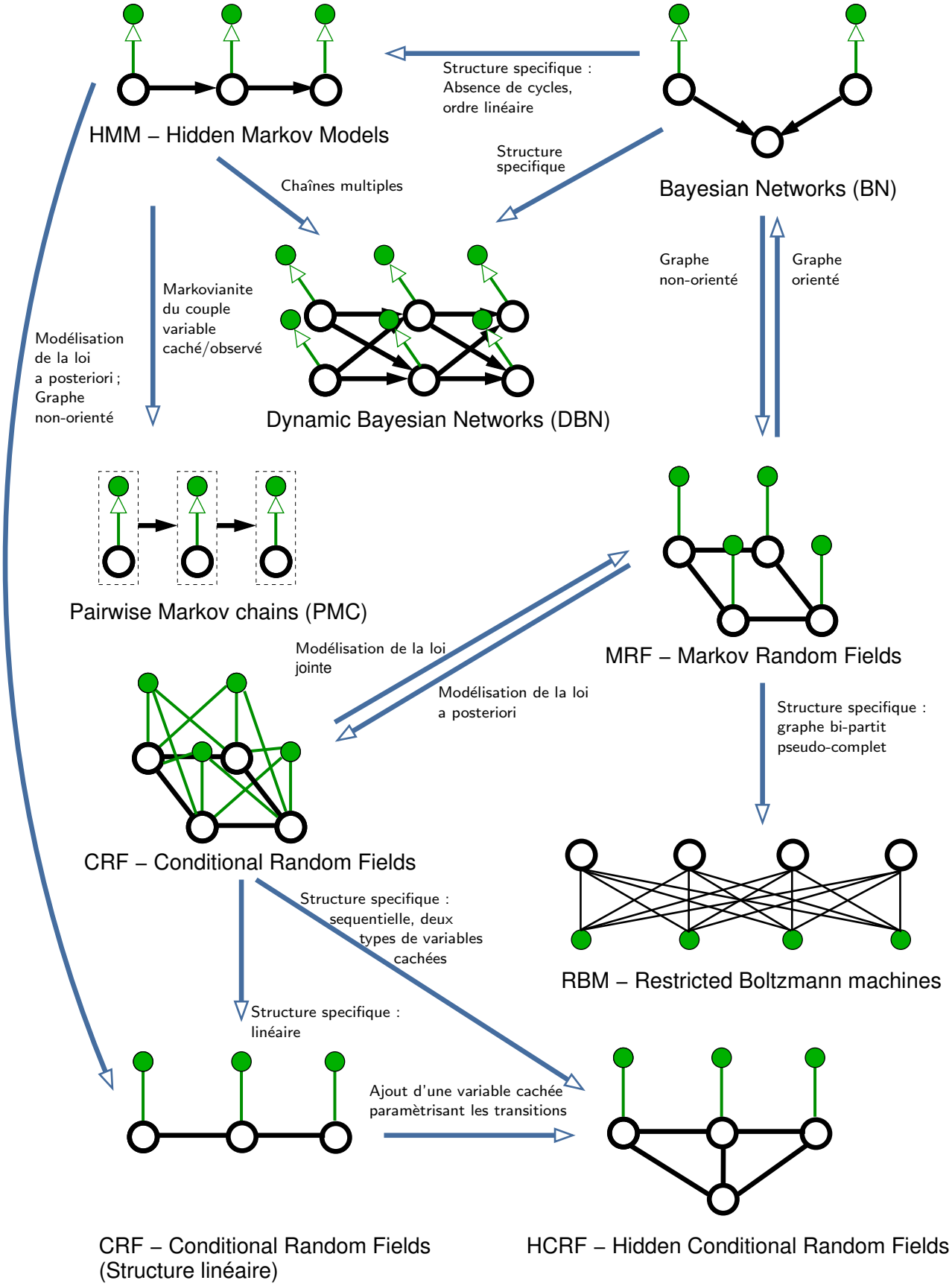


Figure 3.15 – Quelques modèles graphiques probabilistes connus et leurs relations.

.

Les MRF ont été adoptés par la communauté de traitement d'image relativement tôt, surtout grâce à la possibilité de modéliser des structures régulières et irrégulières sans causalité, comme les pixels d'une image [GG84] ou les points d'intérêt ou encore les régions d'un object visuel [Li01]. En conséquence, quand les CRF séquentiels ont été introduits pour modéliser de façon discriminative les séquences d'observations, ils ont été rapidement adaptés à des structures 2D pour la modélisation d'images [KH05] pour remplacer les MRF dans certaines applications.

Plusieurs modèles très répandus ont été conçus explicitement pour des applications spécifiques. Comme seul exemple parmi un grand nombre de cas nous citerons la classification de séquences entières. La résolution de ce problème par HMM ou CRF nécessite la création de plusieurs modèles, un par classe à reconnaitre. Cela présente un inconvénient en soi, puisque les probabilités *a posteriori* entre classes ne sont pas comparables. En outre, les paramètres ne peuvent pas être partagés, même en partie, entre les modèles. Ces faiblesses sont adressées par les HCRF (*Hidden Conditional Random Fields*) [QWM+07] permettant de modéliser plusieurs classes de séquences par le même modèle. Dans ce but, une nouvelle variable cachée unique a été introduite et connectée à toutes les variables cachées classiques du modèle. En conséquence, les distributions conditionnelles gérant les transitions entre états dépendent de deux variables cachées, nécessitant l'estimation d'une matrice de probabilités de transitions à trois dimensions lors de l'apprentissage. Cela n'est pas un inconvénient par rapport aux HMM et CRF, où un ensemble de matrices 2D doit être estimé, une pour chaque modèle.

3.7 Résumé de nos contributions : modèles et applications

Un résumé de nos contributions est donné dans les figures 3.16 et 3.17 en présentant un exemple de graphe de dépendances à gauche et une illustration ou un résultat de l'application dans la colonne de droite. Nous avons traité des problèmes de segmentation d'images et de vidéos à l'aide de MRF et CRF, des problèmes de segmentation de maillage et d'optimisation de maillages avec des CRF, des problèmes d'indexation d'images et de reconnaissance d'objets à l'aide de HMM et de MRF, et des problèmes de reconnaissance d'actions dans la vidéo à l'aide de MRF.

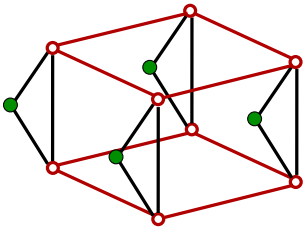
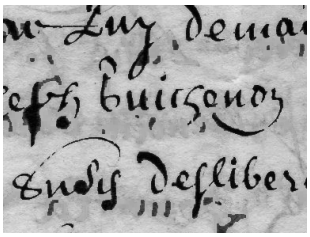
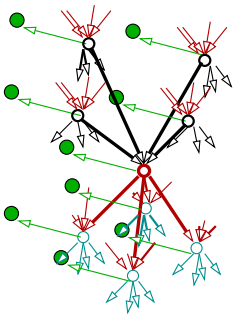
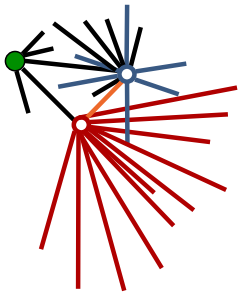
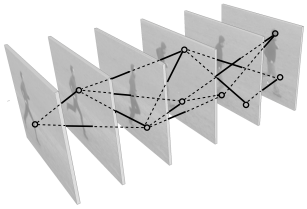
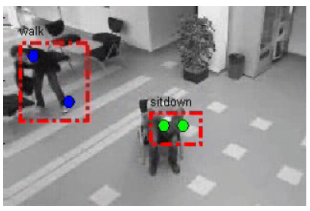
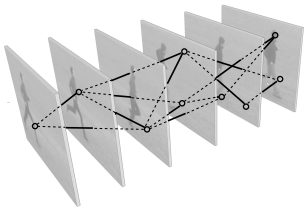
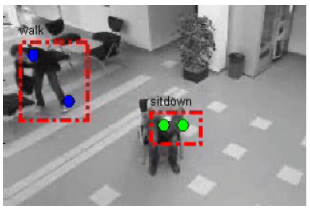
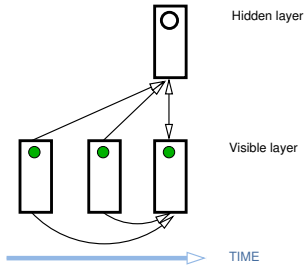
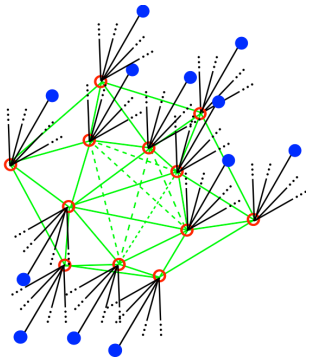
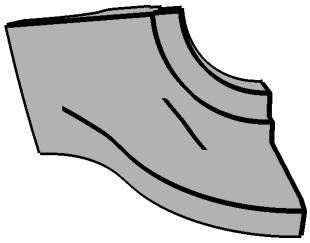
		Restauration d'images de documents : MRF à double couche [Wol10]
		Segmentation et restauration d'images : cube de Markov hierarchique [WG10]
		Segmentation de vidéos : calcul conjoint de la segmentation et de l'estimation de mouvement : MRF spatio-temporel [WJ10]
		Appariement (approché) de graphes sur des données spatio-temporelles : MRF [TWLB10]
		Appariement (exact) de graphes sur des données spatio-temporelles : MRF [CWS12a, CWS12b]

Figure 3.16 – Quelques modèles graphiques développés (gauche) et des applications associées (droite)

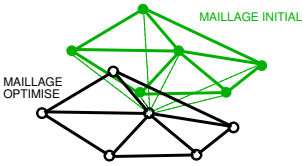


Reconnaissance d'actions : machine de Boltzmann [WT11]

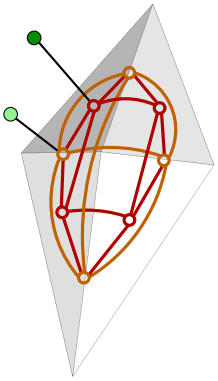


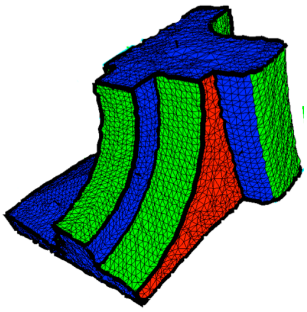


Détection d'arêtes vives dans les maillages : CRF [VWD11]

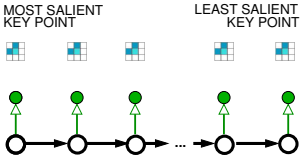


Remaillage/Optimisation de maillages : CRF [VWD12a]





Segmentation/décomposition de maillages en primitives géométriques : CRF [VWD12b]



Classification d'images : HMM [MSW09]

Figure 3.17 – Quelques modèles graphiques développés (gauche) et des applications associées (droite)

Chapitre 4

Modèles (semi)-structurés et appariement de graphes

Dans ce chapitre nous présenterons nos travaux autour de la reconnaissance d'objets dans un sens large, c.à.d. la reconnaissance d'images 2D, d'objets réels ou d'objets dessinés, ainsi que la reconnaissance d'objets spatio-temporels, c.à.d. d'activités dans des séquences vidéo. Le point commun dans cette thématique est la modélisation de tels objets par des modèles structurés ou semi-structurés et leur reconnaissance par des algorithmes adaptés.

Le chapitre est organisé de la manière suivante. Après une description du contexte collaboratif de nos recherches et des projets associés dans la section 4.1, nous plaçons les travaux dans leur contexte scientifique dans la section 4.2, qui donne un aperçu de la modélisation par modèles structurés et modèles semi-structurés. La section 4.3 est une esquisse rapide de nos travaux sur les modèles semi-structurés, c.à.d. les travaux qui ne tiennent compte des relations spatiales et spatio-temporelles de manière incomplète seulement.

Nos travaux principaux seront revendiqués dans les sections suivantes. La section 4.4 donne un état de l'art concis des méthodes d'appariement de graphes et présente le framework par minimisation d'une fonction d'énergie sur lequel s'appuient un grand nombre d'algorithmes, aussi les nôtres. Les sections suivantes proposent des solutions spécifiques pour quelques applications, soit par une modélisation spécifique, soit par une méthode de minimisation spécifique. La section 4.5 propose une méthode de reconnaissance d'objets dessinés. La section 4.6 se focalise sur l'appariement de graphes pour la reconnaissance d'actions dans la vidéo. Le résultat le plus important du chapitre est probablement l'algorithme d'appariement exact de graphes dont les données sont plongées dans l'espace-temps, présenté dans la sous-section 4.6.3.

4.1 Contexte, projets et collaborations

La reconnaissance d'objets est une problématique classique dans le domaine de la vision par ordinateur, elle intervient dans un très grand nombre d'applications. Ce problème fait donc partie de nos activités depuis le début de nos recherches. La figure 4.1 montre les différents travaux sur un axe temporel ensemble avec les projets dans lesquels ils ont été effectués. Les projets ANR Sattic et ANR Canada nous ont incités à étudier les modèles structurés et semi-structurés pour représenter les objets. Le sujet principal de l'ANR Sattic, un projet de l'appel blanc, était l'étude de

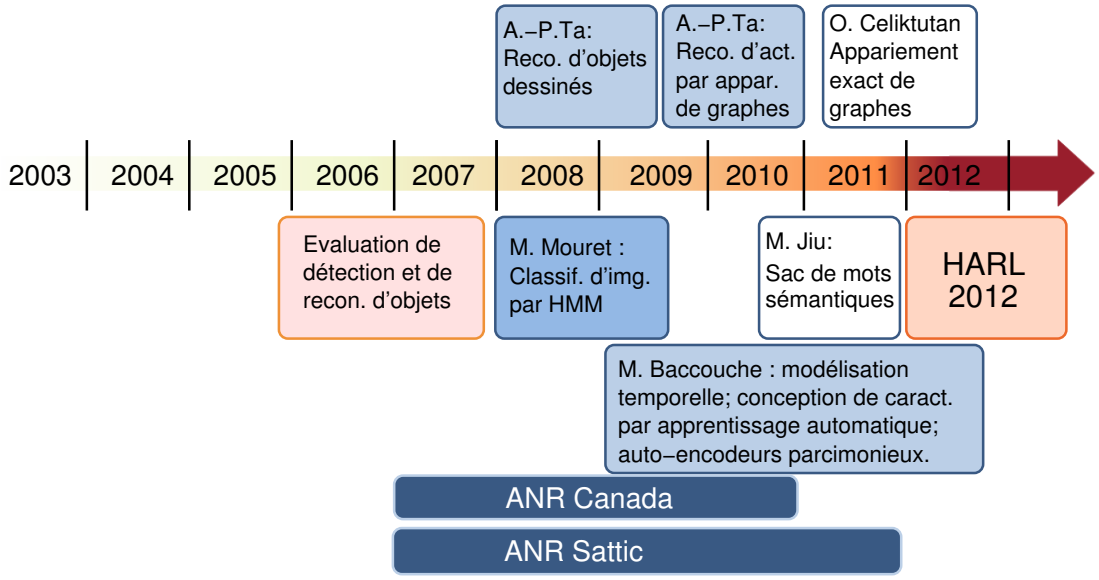


Figure 4.1 – Illustration de l'évolution temporelle des travaux du chapitre 4.

la modélisation d'images par chaînes, par arbres, et, plus généralement, par graphes. Nos premiers travaux avec Christine Solnon, du LIRIS, et l'étudiant Marc Mouret (M2R) s'orientaient dans cette direction [MSW09] — voir aussi la section 4.2.

Nous nous sommes rapidement rendu compte que la modélisation par graphe général, non restreint sur les chaînes et les arbres, apporte des avantages malgré les problèmes de complexité d'appariement. D'un point de vue applicatif, le problème de la reconnaissance d'objets dessinés était l'objectif de la thèse d'Anh Phuong Ta, un étudiant financé par une bourse CIFRE avec l'entreprise d'animation *Pinka Studios*. Nous avons naturellement traité ce sujet avec des algorithmes à base d'appariement par graphes [TWLB09] — voir aussi la section 4.5.

Suite aux problèmes financiers de l'entreprise dus à la crise de l'économie mondiale, et suite à son dépôt de bilan, nous avons légèrement modifié l'aspect applicatif de la thèse d'Anh Phuong tout en gardant son orientation théorique, à savoir les modèles (semi)-structurés et la modélisation par graphes. L'étudiant a passé la deuxième moitié de sa thèse de doctorat sur le sujet applicatif du projet ANR Canada, qui a comme but la reconnaissance d'activités dans les séquences vidéo. Nous avons alors proposé une adaptation du modèle sac de mots en intégrant la prise en compte de relations spatio-temporelles [TWL+10] et une méthode d'appariement de graphes pour cette application [TWLB10].

Encouragé par le succès de l'approche[1], nous avons continué dans cette direction avec Oya Celikutan, une doctorante du laboratoire BUSIM de Bogaziçi University, Istanbul, encadrée par Prof. Bülent Sankur et co-encadrée par moi-même. La différence avec l'approche précédente mise en place par Anh-Phuong Ta réside dans la méthode d'optimisation : au lieu de recourir à une méthode spectrale existante, nous avons conçu une méthode exacte en profitant de certaines propriétés de l'espace 2D+t dans lequel sont plongées les données — voir la section 4.6.3.

Dans le cadre de la thèse de Mingyuan Jiu nous avons repris les modèles de type sac de mots

1. L'article [TWLB10] a été sélectionné comme *Best Paper* du track *Recognition* de la conférence AVSS 2010.

en améliorant leur pouvoir de discrimination. L'idée est d'apprendre le dictionnaire du modèle de manière supervisée et conjointement avec le modèle de prédiction, au lieu de le déterminer de manière non-supervisée dans une étape préliminaire. Une esquisse est donnée dans la section 4.2.

Les travaux sur la reconnaissance d'actions sont liés à d'autres travaux effectués dans la cadre de la thèse de Moez Baccouche, co-encadré par Christophe Garcia et par Atilla Baskurt du LIRIS et encore par Franck Mamalet d'Orange Labs Rennes. Ils se focalisent sur la modélisation séquentielle et sur l'apprentissage automatique de caractéristiques [BMW+10a, BMW+11, BMW+10b] — voir les sections 4.3.3 et 4.3.4.

Toujours dans un contexte de détection et de reconnaissance d'objets dans un sens large, nous avons également proposé une nouvelle méthode d'évaluation d'algorithmes de détection d'objets, en séparant les aspects qualitatifs et quantitatifs et en insistant sur leur dépendance [WJ06]. Notre méthode a été reprise pour évaluer les soumissions des différents participants dans le cadre de plusieurs compétitions scientifiques, à savoir la compétition *ICDAR 2003 robust reading* organisée par Simon Lucas [LPS+05] et *ImageEval 2007* organisé par Pierre-Alain Moellic [WJ07].

Depuis Octobre nous sommes en train d'organiser notre propre compétition scientifique dans le cadre de la conférence internationale *International Conference on Pattern Recognition 2012*. Nommé *Human activities recognition and localization competition — HARL 2012*, elle porte sur la détection et la reconnaissance automatique d'actions humaines [WML+12].

Dans l'avenir, ces travaux se poursuivront dans le cadre de deux thèses débutant en octobre 2012. La première, du doctorant Allaeddine Mihoub, aura comme sujet l'apprentissage de modèles génératifs d'interactions homme-robots, la deuxième, de la doctorante Natalia Neverova, traitera des problèmes de reconnaissance en temps-réel, toujours dans un contexte de robotique mobile.

4.2 Les modèles structurés et semi-structurés

Dans la figure 4.2 nous avons essayé de donner une illustration, forcement incomplète, de quelques modèles structurés et semi-structurés (MSSS) souvent utilisés pour représenter et pour détecter et reconnaitre les objets et les actions. Nous les avons placés sur un axe passant d'une invariance maximale à gauche vers un pouvoir de discrimination maximal à la droite. La notion d'« invariance » traitée ici concerne la notion classiquement considérée en vision par ordinateur, c.à.d. l'invariance par rapport aux transformations telles que les changements d'éclairage, les changements d'échelle, les rotations, les mouvements articulés, les changements topologiques etc.

Il va sans dire que cet axe entre invariance et pouvoir de discrimination n'est pas une manière à cent pour cent exacte de classer les travaux de l'état de l'art : des avancées de la recherche scientifique nous permettent d'augmenter les deux critères à la fois. Néanmoins, très souvent une invariance accrue s'obtient seulement au détriment d'un plus faible pouvoir de discrimination.

Dans la figure, les MSSS ciblés sont complétés et délimités par ce que nous considérons comme deux cas extrêmes : par les histogrammes de couleur [SB91] comme le modèle le plus invariant, à notre connaissance, et par l'estimation par RANSAC d'une transformée globale comme le modèle le moins invariant. Pour des raisons liées aux aspects applicatifs, ces deux modèles sont utilisés pour les images, mais pas pour les actions. Les histogrammes calculés sur les couleurs de pixels sont des modèles assez anciens mais toujours très utilisés pour la classification d'images. Comme ils ne permettent pas de séparer le mouvement de l'apparence, leur utilisation directe pour les séquences vidéos est difficile. De manière similaire, et dans l'autre extrémité de notre axe, les appariements

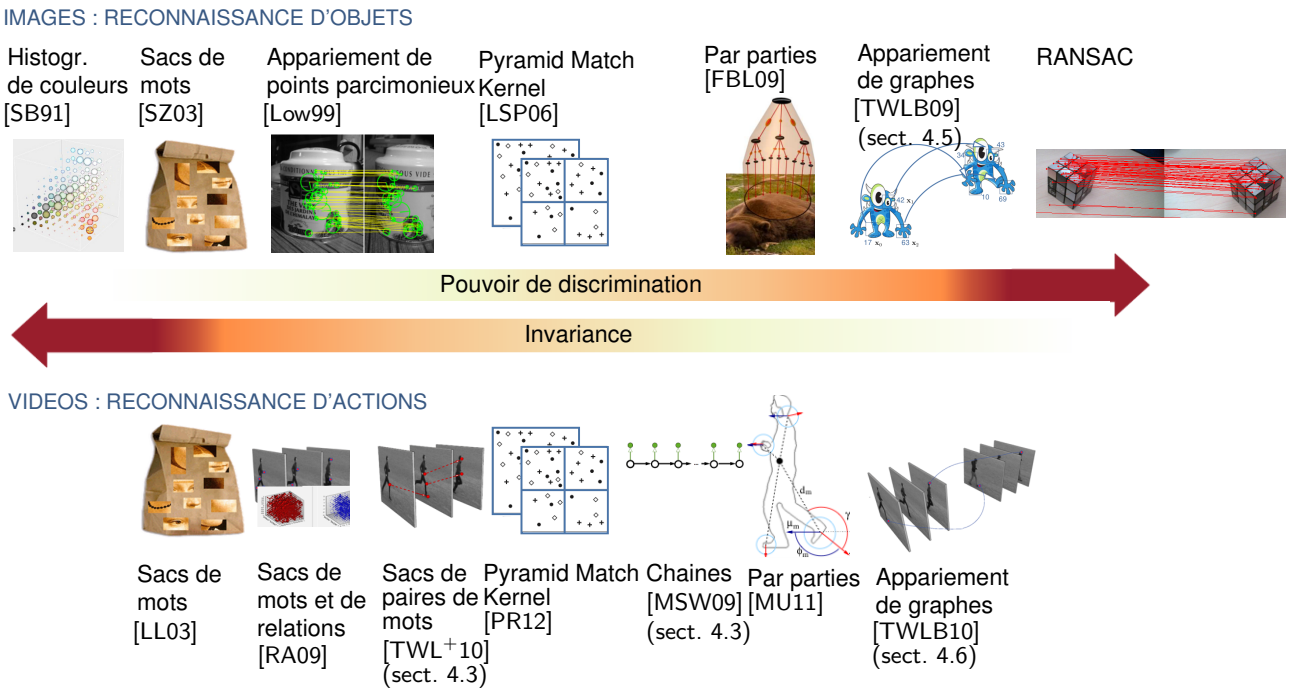


Figure 4.2 – Les différents compromis entre invariance maximale (à gauche) et pouvoir de discrimination maximale (à droite). En haut : modélisation d'images. En bas : modélisation de vidéos. Illustration du sac de mots prise de Robert Fergus (`http://people.csail.mit.edu/fergus/iccv2005/bagwords.html`), illustration du RANSAC prise de `http://srand2.blogspot.com`, l'illustration de l'appariement de points de [RL+07]. Les illustrations pour les méthodes suivantes ont été reproduites à partir des articles respectifs : [FBL09, MU11].

de nuages de points par transformée globale permettent la reconnaissance robuste d'objets rigides. Les corps humains étant articulés, il est difficile d'apparier deux corps ou deux actions avec une telle méthode.

Grâce à leur robustesse vis à vis d'occultations, les représentations par points parcimonieux (ou points d'intérêt) ont eu un grand succès dans la communauté. Il s'agit d'extraire un sous-ensemble saillant de points d'une image ou d'une vidéo, et de le compléter par des caractéristiques extraites d'une zone autour de chaque point. La description ainsi obtenue est structurelle dans la manière où elle consiste en un ensemble de points dont les relations spatiales ou spatio-temporelles sont souvent aussi importantes que les caractéristiques d'apparence associées. Leur utilisation directe par des méthodes basées sur l'apprentissage est difficile, car la plupart des classifieurs nécessitent une description plongée dans un espace vectoriel.

Pour pallier à un problème similaire, les modèles *Sacs de mots* (*Bags of words*) (BoW) ont été introduits dans la communauté de recherche de textes. Il s'agit de modéliser un document textuel par un histogramme représentant la fréquence des mots [SAY75]. Dans le contexte de vision par ordinateur, les mots textuels ont été remplacés par des mots visuels, c'est à dire, des motifs visuels locaux discriminants et fréquents, donnant ainsi naissance au modèle BoW [SZ03]. Ce modèle nécessite quatre étapes de base : (i) la détection de primitives locales, (ii) l'extraction de caractéristiques locales (iii) la création d'un dictionnaire visuel (iv) la représentation d'une

image ou d'une vidéo comme un histogramme. Les dictionnaires visuels sont typiquement obtenus par un clustering non supervisé des caractéristiques sur l'ensemble d'apprentissage, souvent par l'algorithme k-moyennes. Chaque cluster donne un mot et chaque image ou vidéo peut alors être transformée en un histogramme.

Le concept a également été étendu aux vidéos s'appuyant sur des détecteurs de points saillants de type espace-temps [LL03, DRCB05, SAS07, LMSR08]. De nombreuses extensions ont été introduites avec l'objectif d'augmenter le pouvoir de discrimination de la méthode sans trop heurter l'invariance. A titre d'exemples nous citerons [SDNFF08], où la corrélation des modèles est mesurée et des corrélogrammes espace-temps sont construits ; [NFF07], où la distribution spatio-temporelle des caractéristiques est modélisée de manière probabiliste pour chaque mot visuel ; [LS08], où les points d'intérêt sont regroupés en maximisant l'information mutuelle ; et [RA09], où des histogrammes 3d sont construits à partir de paires de points. Deux dimensions de chaque histogramme correspondent aux clusters des deux points des paires, et la troisième dimension correspond à une discrétisation des relations spatio-temporelles des paires (proche, loin, recouvrement, avant, après etc.). Cette approche est montrée dans la figure 4.2 pour représenter toutes les approches avec un niveau moyen de prise en compte des relations spatiales : ni une prise en compte totale, comme pour le cas de l'appariement de graphes, ni aucune prise en compte, comme pour les BoW. Dans [BB12], des histogrammes similaires, par paires, sont créés pour chaque point de la vidéo, les paires étant extraites du voisinage du point. Les histogrammes sont ensuite regroupés par *clustering* pour la construction de modèles de type sac de mots.

Une extension différente des BoW sont les modèles de type *Pyramid Match Kernel* [GD05, LSP06]. Une grille hiérarchique de histogrammes est créée et une mesure d'intersection est calculée couche par couche et cellule par cellule. Cela permet d'introduire les relations entre les données de manière hiérarchique. La différence entre [GD05] et [LSP06] réside dans la manière comment ce principe est appliqué. Dans [GD05], la hiérarchie est créée dans l'espace des caractéristiques, les relations spatiales sont ignorées. Dans [LSP06], l'appariement hiérarchique est organisé de manière spatiale, l'espace de caractéristiques est traité par *clustering* comme un modèle de type BoW classique. Ces modèles ont récemment été étendus pour tenir compte des aspects temporels [PR12].

Les approches par parties (*parts based models*), introduites assez tôt [FE73], modélisent un objet en le décomposant en parties stockées ensemble avec leurs positions spatiales. Les modèles graphiques probabilistes sont fréquemment utilisés dans ce contexte [FH05, WS06, FGMR10]. La création d'une décomposition hiérarchique de parties en sous parties permet le partage (partiel) d'un dictionnaire pour plusieurs objets et une recherche efficace [FBL09]. Des approches similaires ont été introduites pour la reconnaissance d'actions [MU11]. Dans [RBBT11], une action est décomposé en acteurs physiques, sous-évènements et contraintes. Les actions sont modélisées par une grammaire et reconnues par un algorithme probabiliste.

Une modélisation complémentaire décompose une entité en attributs, où un attribut est une propriété associée à un verbe, par exemple « s'asseoir », « être débout » etc. [YJK+11, LKS11]. Ce gendre de décomposition est généralement combinée avec une décomposition en parties.

L'intégration des aspects temporels permet de tenir compte des spécificités des applications vidéo. A titre d'exemples nous citons les BoW temporels appariés à l'aide d'une version temporelle et hiérarchique de la *Earch Mover's Distance* [XC08], une distance dont la version originale a été conçue pour apparier, de manière robuste, des histogrammes décrivant des formes visuelles [RTG00]. Dans [KB12], les primitives locales à l'origine du modèle sont des trajectoires de points

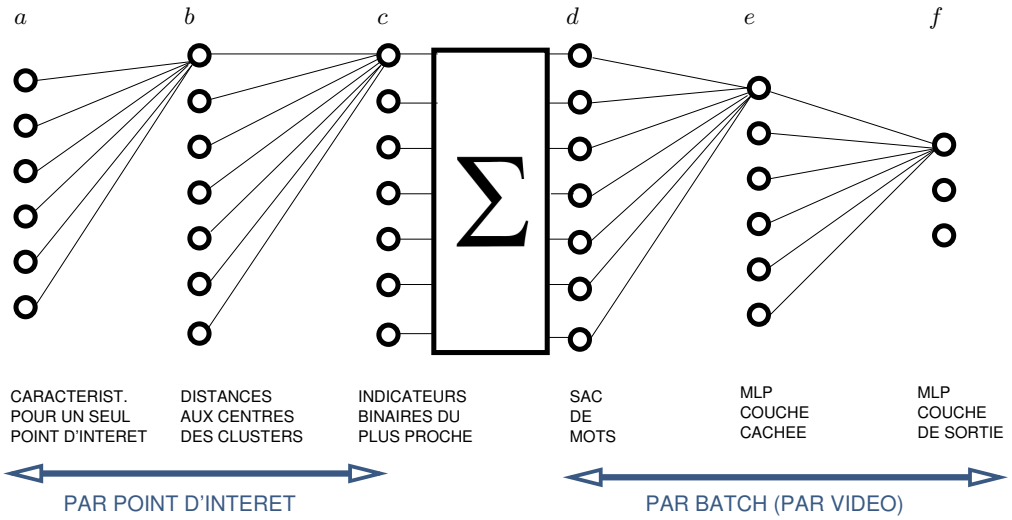


Figure 4.3 – Sacs de mots sémantiques : apprendre conjointement le dictionnaire et le modèle de prédiction.

d'intérêts suivis avec un filtre de Kalman. Un descripteur est calculé sur chaque trajectoire, comprenant des caractéristiques d'apparence et une partie calculée sur les positions 2D. Des *mots de vidéo* sont extraits par clustering et regroupement.

4.3 Nos contributions à la modélisation par modèles semi-structurés

Avant de passer à la modélisation par graphes et à l'appariement de graphes dans la section 4.4 et les sections suivantes, le sujet principal de ce chapitre, nous présenterons dans cette section quelques modèles semi-structurés que nous avons développés dans le cadre de nos recherches. Les applications décrites concernent à la fois l'analyse d'images et de vidéos.

4.3.1 Dictionnaires sémantiques

Les premiers travaux dans le cadre de la thèse de Mingyuan Jiu, co-encadré par Atilla Baskurt et moi-même, étaient consacrés à l'amélioration de l'approche classique BoW, tout en gardant sa simplicité et son faible coût de calcul [JWGB12]. Ces travaux ont été effectués en collaboration avec Christophe Garcia. Il s'agit d'augmenter le pouvoir de discrimination du dictionnaire de l'approche en l'apprenant de manière supervisée. En effet, dans l'approche classique, les étiquettes de classe des données d'apprentissage sont utilisées pour l'apprentissage du modèle de prédiction uniquement. Le dictionnaire est classiquement créé de manière non-supervisée sans prise en compte des étiquettes. Notre amélioration consiste à intégrer les deux étapes, à savoir la création du dictionnaire et l'apprentissage du modèle de prédiction, en un seul modèle de type réseau neuronal.

La figure 4.3 montre l'architecture du réseau implémentant l'approche. Contrairement à un modèle neuronal classique, la classification procède de manière itérative en intégrant plusieurs vecteurs de caractéristiques avant le calcul du résultat. Chaque vecteur d'entrée correspond au

descripteur associé à un point d'intérêt. La première partie du modèle réalise une projection du vecteur sur un dictionnaire, dont les centres de cluster sont codés dans les poids d'une couche du modèle, donnant lieu à un vecteur binaire indiquant le centre le plus proche du vecteur d'entrée. La deuxième partie produit un BoW en intégrant tous les indicateurs et puis le passe à quelques couches de type MLP (*multilayer perceptron*) pour les classifier.

Nous avons proposé deux méthodes d'apprentissage conjoint du dictionnaire et des poids de la partie MLP du modèle. La première consiste à rétro-propager le gradient de l'erreur de classification comme pour les MLP traditionnels. Deux adaptations sont toutefois nécessaires. D'une part, l'intégrateur entre les deux parties nécessite une distribution de l'erreur sur les histogrammes (couche « d » dans la figure 4.3) sur la partie individuelle (couche « c » dans la figure 4.3), par exemple de manière uniforme. D'autre part, l'indicateur du plus proche centre, non dérivable dans sa version classique, doit être approximé par un *softmin* afin de pouvoir rétro-propager l'erreur sur cette partie.

La deuxième méthode d'apprentissage consiste à bénéficier de nos connaissances sur la signification des couches intermédiaires du réseau. L'erreur sur la couche correspondant aux BoW est donc correctement interprétée comme un ensemble d'affectations incorrectes de descripteurs (de points) aux mots visuels (les centres de clusters). Les centres sont changés à l'aide d'un diagramme de Voronoi calculé sur ces dernières, comme il est illustré dans la figure 4.4. Dans la figure, le point P est transféré d'une cellule de Voronoi dans une autre afin de « corriger » l'erreur. Ensuite, les centres sont recalculés comme étant les moyennes des vecteurs compris dans les cellules de Voronoi respectives.

L'avantage de l'approche proposée par rapport aux formalisme de BoW classique réside dans la construction d'un dictionnaire d'un pouvoir de discrimination plus important. Cela permet d'augmenter la performance de classification, ou, alternativement, de diminuer la taille de l'histogramme tout en gardant les performances du système.

4.3.2 Intégration de la géométrie espace-temps dans le formalisme BoW

Dans le cadre de la thèse d'Anh-Phuong Ta, co-encadré par Guillaume Lavoué, Atilla Baskurt et moi-même, et dans un contexte de reconnaissance d'actions, nous avons proposé une méthode de type BoW améliorée [TWL+10] qui consiste à traiter des paires de points au lieu de points simples. La différence avec l'algorithme de Ryoo et al. [RA09], qui procède également par paires, réside dans la manière avec laquelle les relations spatio-temporelles entre les points d'une paire sont prises en compte. Dans [RA09], le dictionnaire est de la même nature que celui de l'approche BoW classique. Les BoW sont des histogrammes 3D, où deux dimensions correspondent aux mots clés des deux points respectifs, et la troisième dimension correspond à une discrétisation de leurs relations spatiales en un ensemble de possibilités (*proche en espace, loin en espace, proche en temps, loin en temps, recouvrement* etc.). La prise en compte de la géométrie espace-temps est donc relativement faible.

Notre approche, en revanche, adapte la création du dictionnaire afin de mieux prendre en compte les relations spatio-temporelles entre les points d'une paire. Plus précisément, deux dictionnaires sont créés : un dictionnaire d'apparence calculé à partir des descripteurs associés aux deux points de la paire, et un dictionnaire de géométrie espace-temps calculé à partir du vecteur trois-dimensionnel entre les deux points (voir la figure 4.5). La création du dictionnaire géométrique procède par *clustering* de type k-moyennes des vecteurs trois-dimensionnels associés aux

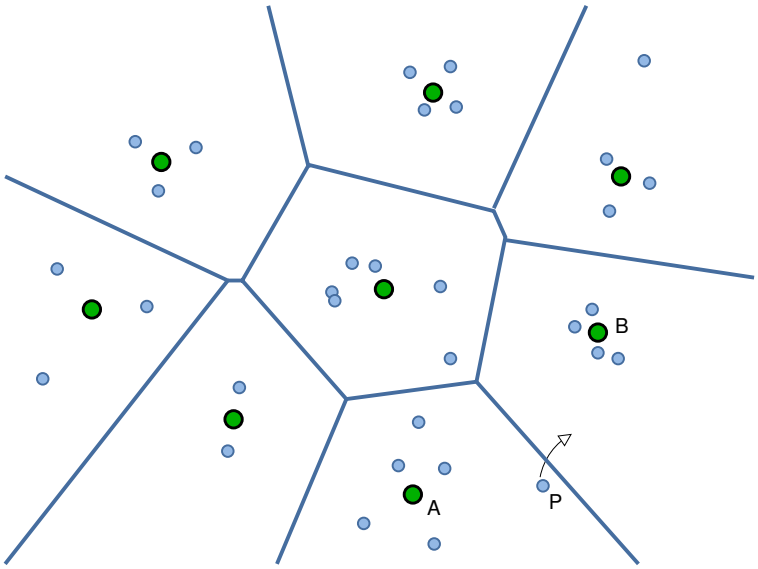
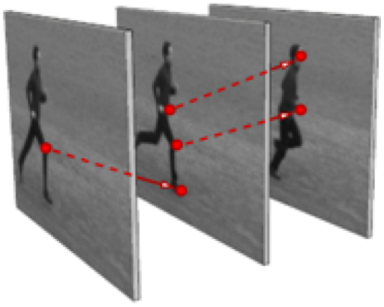


Figure 4.4 – Apprentissage d'un dictionnaire sémantique : illustration du changement de cluster, par un point, à l'aide d'un diagramme de Voronoi calculé pour l'espace de caractéristiques. Pour des raisons de simplicité, ici l'espace de hautes dimensions est illustré par un espace 2D.

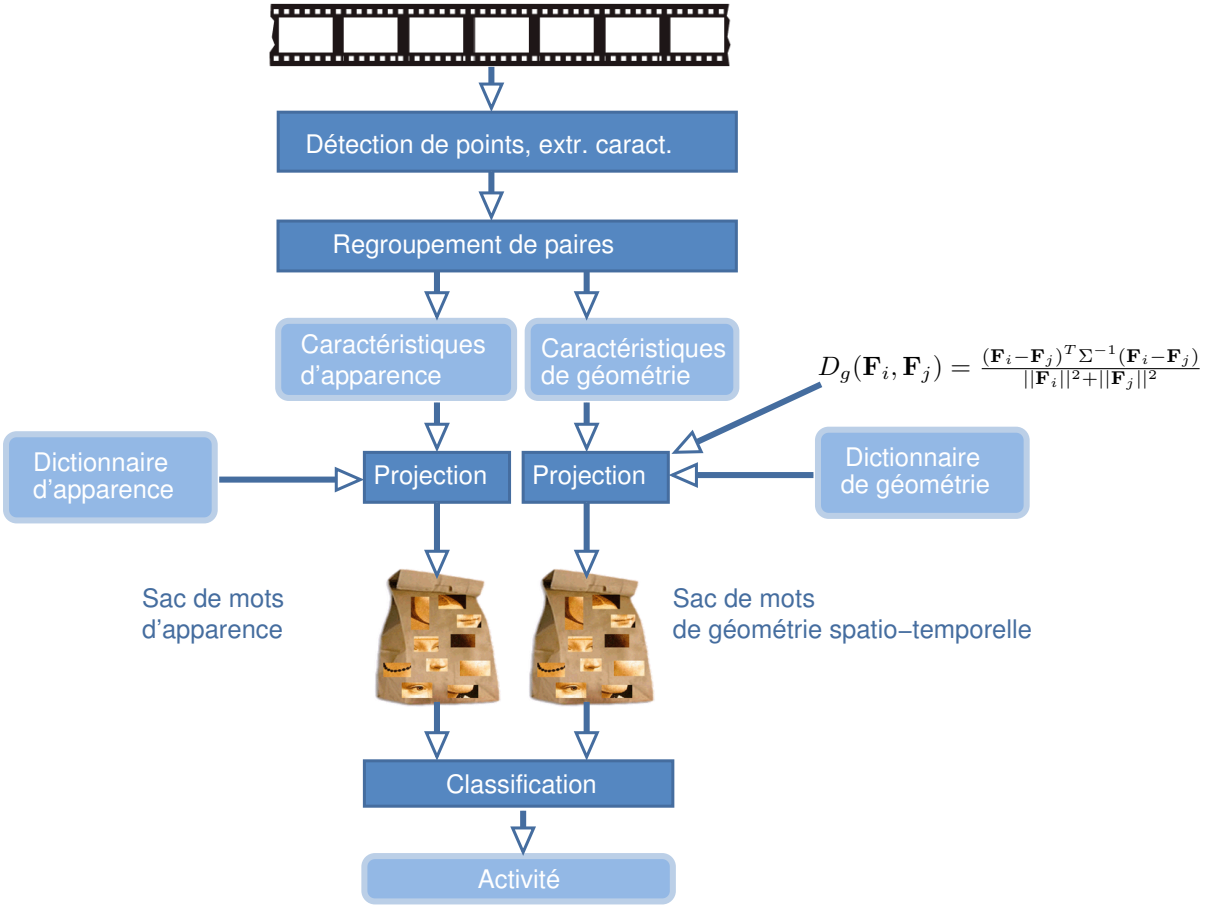
pairs. Cela nécessite une distance entre deux vecteurs $\mathbf{f}_i$ et $\mathbf{f}_j$, que nous avons définie comme une distance Euclidienne normalisée dans un espace déformé par une transformation linéaire, similaire à une distance de Mahalanobis :

$$D(\mathbf{f}_i, \mathbf{f}_j) = \frac{(\mathbf{f}_i - \mathbf{f}_j)^T \Sigma^{-1} (\mathbf{f}_i - \mathbf{f}_j)}{||\mathbf{f}_i|| + ||\mathbf{f}_j||} \tag{4.1}$$

Ici Σ est une matrice diagonale dont les valeurs servent à pondérer les trois dimensions de l'espace-temps. La classification se fait par un classifieur standard, en l'occurrence un SVM, sur un vecteur composé de deux histogrammes : un histogramme construit à partir du dictionnaire d'apparence, et un histogramme construit à partir du dictionnaire de géométrie (voir la figure 4.5b). L'avantage de cette formulation est la richesse du descripteur géométrique. Il est tout à fait possible de se servir d'un seul dictionnaire au lieu de la combinaison des deux. Si seulement le descripteur d'apparence est utilisé, on revient à la méthode classique du BoW comme proposée dans [LL03]. Par contre, il s'avère qu'il est beaucoup plus intéressant de recourir à une approche inverse, c.à.d. de se servir de la géométrie uniquement sans l'apparence. Malgré la légère différence en performance en sa défaveur illustrée dans la table 4.1a, l'intérêt de l'approche se confirme dans un cas où la base d'apprentissage n'est pas complètement représentative des données de test. Des résultats d'expériences d'apprentissage croisé sont montrés dans la figure 4.1b sur deux bases de vidéos différentes : en se servant des 3 classes communes des deux bases de vidéos KTH [SLC04] et Weizmann [GBS+07], des expériences ont été menées en apprenant sur une base et en testant sur l'autre. Nous pouvons remarquer que les caractéristiques de géométrie espace-temps sont bien plus robustes à ce genre de changement de base, ce qui les rend beaucoup plus intéressantes pour des applications réalistes.



(a)



(b)

Figure 4.5 – Sac de mots de paires : regroupement des points d'intérêts en paires par critère de proximité. (a) les vecteurs connectant deux points d'une paire constituent une information supplémentaire aux descripteurs d'apparence ; (b) le schéma de la méthode classification.

Dictionnaire	Base pour l'apprentissage	Base pour le test	N.d. classes	Performance
Apparence	KTH	KTH	6	90.5
Géométrie	KTH	KTH	6	88.2
App.+Géom.	KTH	KTH	6	93.0

(a)

Dictionnaire	Base pour l'apprentissage	Base pour le test	N.d. classes	Performance
Apparence	KTH	WM	3	40.0
	WM	KTH	3	48.2
Géométrie	KTH	WM	3	70.0
	WM	KTH	3	83.3

(b)

Table 4.1 – Expériences avec une approche de type BoW à deux dictionnaires (apparence et géométrie). (a) Résultat sur la base KTH [SLC04] ; (b) Résultats d'apprentissage croisée entre KTH et Weizmann [GBS+07].

4.3.3 Évolution temporelle de modèles de type sac de mots

Dans le cadre de la thèse de Moez Baccouche, nous nous sommes penchés sur le problème de la classification automatique de séquences vidéo d'actions de sport. Les applications ciblées étaient de type « magnétoscope numérique intelligent » permettant un accès rapide aux scènes clé d'un évènement, telles que les tirs au but d'un match de football. Nous avons proposé une approche intégrant une grande quantité d'informations disponibles dans le signal, à savoir

- des informations d'apparence extraites frame par frame ;
- des informations sur le mouvement de la caméra entre deux frames, qui sont liées aux intentions du metteur en scène de la vidéo ;
- les évolutions temporelles des deux derniers types d'information.

En effet, les vidéos de ce type sont généralement mises en scène par un opérateur humain qui injecte une information supplémentaire sur le contenu par le biais du mouvement de la caméra. Un tir au but, par exemple, ou un but même, sont souvent accompagnés d'un mouvement horizontal rapide ou d'un zoom. Dans nos travaux nous avons montré que cette information est riche et complémentaire aux informations classiques [BMW+10a].

Dans notre méthode, les caractéristiques d'apparence sont modélisées par un modèle de type « sac de mots » (BoW) construit à partir des points d'intérêt de type SIFT [Low99] et à partir des descripteurs associés. Le mouvement de la caméra est approximé par un mouvement affine et dominant estimé à partir d'une correspondance entre les points SIFT de deux frames consécutives à l'aide de l'algorithme RANSAC [FB81].

La classification de l'évolution temporelle de ces caractéristiques extraites est gérée dynamiquement par un modèle neuronal, basé sur les réseaux de neurones récurrents à large « mémoire court-terme » (LSTM) [HS97]. Les expérimentations faites sur la base « MICCSoccer-Actions-4 » montrent que l'approche neuronale de classification permet d'obtenir des résultats supérieurs à l'état de l'art sur cette base, c.à.d. un modèle de BoW pour tout une vidéo classifiée par SVM

(a)

Méthode		Taux de classification
[BBBS09]	BoW + k-ppv	53%
	BoW + SVM	73%
[BMW+10a]	BoW + LSTM	76%
	Mouvement dominant + LSTM	77%
	BoW + mouvement dominant + LSTM	92%

(b)

Figure 4.6 – (a) les 4 classes d'actions collectives de la base *MICCSoccer-Actions-4* [BBBS09] ; (b) résultats de classification par [BBBS09] et par notre approche [BMW+10a].

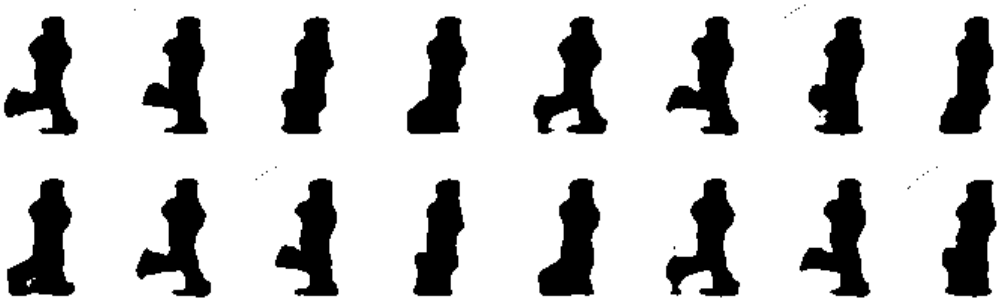
[BBBS09]. La figure 4.6 montre que l'utilisation unique du mouvement dominant est aussi performant que l'information visuelle modélisée par les BoW, donnant des taux de classification entre 76% et 77%. La combinaison des caractéristiques permet une amélioration significative, à savoir un taux de bonne classification de 92%.

4.3.4 Modélisation séquentielle d'une silhouette humaine et conception de caractéristiques par apprentissage

Nous avons proposé plusieurs approches pour la reconnaissance d'actions humaines individuelles, décrites dans ce chapitre. La majorité de nos travaux sur ce sujet passent par une modélisation à l'aide de points d'intérêt spatio-temporels. Ici nous discuterons brièvement une approche alternative, consistant i) à créer une séquence de silhouettes binaires à l'aide d'une segmentation fond/forme ; ii) à extraire des caractéristiques de forme de chaque silhouette ; iii) et à modéliser l'évolution temporelle des descripteurs par un modèle de séquences, tel qu'un HMM, un CRF etc.

En collaboration avec Graham Taylor de l'Université de Guelph [2], et avec Jean-Michel Jolion du LIRIS, nous avons montré qu'un mouvement humain peut être modélisé et reproduit artificiellement à l'aide d'un modèle graphique probabiliste génératif de type « machine de Boltzmann conditionnelle restreinte » (CRBM) [WT11]. Contrairement aux HMM et aux CRF, l'état caché d'un tel modèle est vectoriel, ce qui permet de modéliser des interactions plus riches [THR07]. La figure 4.7a montre quelques frames consécutives produites artificiellement par un CRBM dont les paramètres ont été appris sur des silhouettes extraites de quelques vidéos d'apprentissage. Les observations fournies comme entrées au modèle sont les moments de Zernike extraits sur chaque silhouette. En regardant une grande quantité de frames que nous avons échantillonnées (seulement 16 sont montrées dans la figure 4.7a), nous avons remarqué que le modèle est capable de produire

2. Graham Taylor était avec l'Université de New York au moment de ces travaux.



(a)

(b)

Figure 4.7 – Modélisation par une « machine de Boltzmann conditionnelle restreinte » : (a) une séquence de frames créées artificiellement à partir du modèle de type ; (b) résultats de reconnaissance sur les vidéos acquises par l'École de Mines de Douai dans la cadre du projet ANR Canada.

un vrai mouvement sans répétitions exactes. Il ne s'agit donc pas d'un simple processus de stockage dans un tableau.

Le modèle peut également servir à la reconnaissance en passant les états cachées du CRBM dans un SVM pour les classifier. La figure 4.7b montre quelques résultats de reconnaissance d'activités individuelles obtenues sur une base vidéos acquises par l'Ecole de Mines de Douai dans la cadre du projet ANR Canada.

Dans le cadre de la thèse de Moez Baccouche, et toujours dans le contexte de la reconnaissance d'activités humaines individuelles, nous avons cherché à rendre la conception de modèles de classification temporelle plus automatique [BMW+12a, BMW+12b]. Contrairement à la méthodologie dominante, qui s'appuie sur des caractéristiques définies manuellement et de manière optimale pour un problème donné, notre méthode apprend, de manière entièrement automatique, une représentation locale et invariante à la translation sans aucune utilisation de connaissances *a priori* [BMW+12b].

Inspirée par la méthode de Ranzato et al. [RHBL07] pour la reconnaissance d'objets 2D, notre approche décompose une vidéo en blocs spatio-temporels, dont la taille spatiale correspond à la boîte englobante normalisée de la silhouette humaine. Chaque bloc est lui-même décomposé en petits cubes spatio-temporelles, l'entité sur laquelle l'apprentissage est effectué. Un auto-encodeur basé sur un réseau de neurones convolutionnel est entrainé à projeter les cubes d'entrées en un code parcimonieux et à reproduire ces entrées. Notre approche apprend les paramètres de l'auto-encodeur en minimisant une fonction d'énergie globale basée sur l'erreur de reconstruction.

A la différence de [RHBL07], une nouvelle variable cachée t_i décrit une translation 3D du cube i par rapport à sa position initiale. La reconstruction remplace donc chaque cube i par un autre cube dans un voisinage spatio-temporel. Les variables t_i étant déterminées avec les autres paramètres du système en minimisant l'erreur de reconstruction, chaque cube est donc remplacé par le cube du voisinage dont la reconstruction est optimale par rapport aux paramètres actuels. Cela rend le représentation invariante par rapport aux translations, et en conséquence la variabilité des caractéristiques apprises par le système est augmentée.

L'évolution temporelle des caractéristiques éparses obtenues sur tous les cubes d'un bloc est apprise par un réseau de neurones récurrent à large « mémoire court-terme (LSTM — voir aussi la section 4.3.3) afin de classer chaque séquence. Cette méthode étant indépendante de l'application, nous avons montré qu'elle donne des excellentes résultats sur les deux applications suivantes :

- les performances sur le problème de la reconnaissance d'actions humaines ont été testées en appliquant la méthode à la base connue KTH [SLC04] — voir également la section 4.3.2 à la page 79 et la section 4.6.4 à la page 107 pour d'autres expériences sur cette base. Le taux de classification de 95.83% obtenu par notre méthode se trouve parmi les meilleurs du domaine, tout en étant basé sur un système automatique ;
- les performances sur le problème de la reconnaissance d'expressions faciales ont été testées sur la base GEMEP-FERA. Sur cette base, nous obtenons actuellement la meilleure performance du domaine [BMW+12b], comme illustré dans la figure 4.2b.

4.3.5 Classification d'images par HMM

Il est crucial de trouver un bon compris entre invariance et pouvoir de discrimination. Comme nous l'avons vu, le compromis en question peut largement dépendre de l'application en question. Pour certaines applications, comme la reconnaissance d'objets rigides, le pouvoir de discrimination peut

Méthode	Perf.
Notre méth. [BMW$^+$12b]	95.83

(voir la figure 4.6, page 109, pour une comparaison avec l'état de l'art)

(a)

Méthode	Ind.	Dep.	Perf.
Notre méth. [BMW$^+$12b]	**80.75**	**98.46**	**87.57**
Yang and Bhanu [YB11]	75.23	96.18	83.78
Tariq *et al.*[TLL$^+$11]	65.50	100.0	79.80
Littlewort *et al.*[LWW$^+$11]	71.40	83.70	76.10
Dhall *et al.*[DAGG11]	64.80	88.70	73.40
Meng *et al.*[MRPBB11]	60.90	83.70	70.30
Valstar *et al.*[VJM$^+$11]	44.00	73.00	56.00

(b)

Table 4.2 – Auto-encodeurs parcimonieux : (a) résultat sur la base KTH pour une application de reconnaissance d'actions — voir également la figure 4.6, page 109, pour une comparaison avec l'état de l'art ; (b) résultats sur la base GEMEP-FERA pour une application à la reconnaissance d'émotions. Ind.=indépendant de la personne ; Dep.=dépendant de la personne ; Perf=performance totale.

être maximisé au détriment de l'invariance. La situation est très différente pour la classification d'images, le cas traité dans cette section, où il convient de maximiser l'invariance de la représentation. Il s'agit d'inférer, à partir d'une image, de quelle classe il s'agit, sachant que les classes sont souvent définies de manière sémantique. A titre d'exemple, il peut s'agir de classes comme *portraits, couchés de soleil, plages, grattes ciel, voitures, bébés* etc.

Il est donc important de maximiser l'invariance, et pour cette raison les premiers travaux sur ces applications modélisaient une image par un histogramme 3D des couleurs des pixels [SB91]. Améliorer ces résultats nécessitera une augmentation du pouvoir de discrimination sans heurter trop à l'invariance.

En collaboration avec Christine Solnon du LIRIS et Marc Mouret, nous avons proposé une structuration des points d'intérêt d'une image en chaînes, donc en séquences [MSW09]. Pour maximiser l'invariance, nous avons écarté les informations spatiales de la définition de l'ordre des points. Le critère retenu était la saillance des points d'intérêt, donc la mesure calculée par les détecteurs de points d'intérêt pour définir son « intérêt ». Ce genre de modélisation par chaînes a déjà été utilisée dans d'autres travaux [RLJS05]. Par contre, les distances classiques supposant des interactions représentatives entres les caractéristiques des points, tel que la distance de Levenstein [Lev66], ne sont pas pertinentes dans ce contexte. Une combinaison avec des distances de histogrammes a été tentée dans [SJ07].

Pour pallier a ce problème nous avons proposé une modélisation par HMM, introduisant des états cachés dans le modèle. Au lieu de modéliser directement les interactions entre les caractéristiques, les interactions sont gérées entre des paires d'états cachés consécutifs et entre chaque état caché et son descripteur associé — voir aussi le chapitre 3, section 3.2.1. La figure 4.8 donne une illustration de la méthode. Un ensemble de points d'intérêt est sélectionné à l'aide du détecteur proposé par Brès et Jolion [BJ99], qui délivre en même temps également une valeur de saillance pour chaque point [3]. Une chaîne est formée à partir d'un ordre imposé par cette saillance, et un

3. Dans le cas spécifique du détecteur [BJ99], la saillance est définie comme un critère de contraste local extrait à l'aide d'une pyramide. En général, la notion de saillance n'est pas définie de manière exacte, sa discussion dépasse le cadre de ce mémoire.

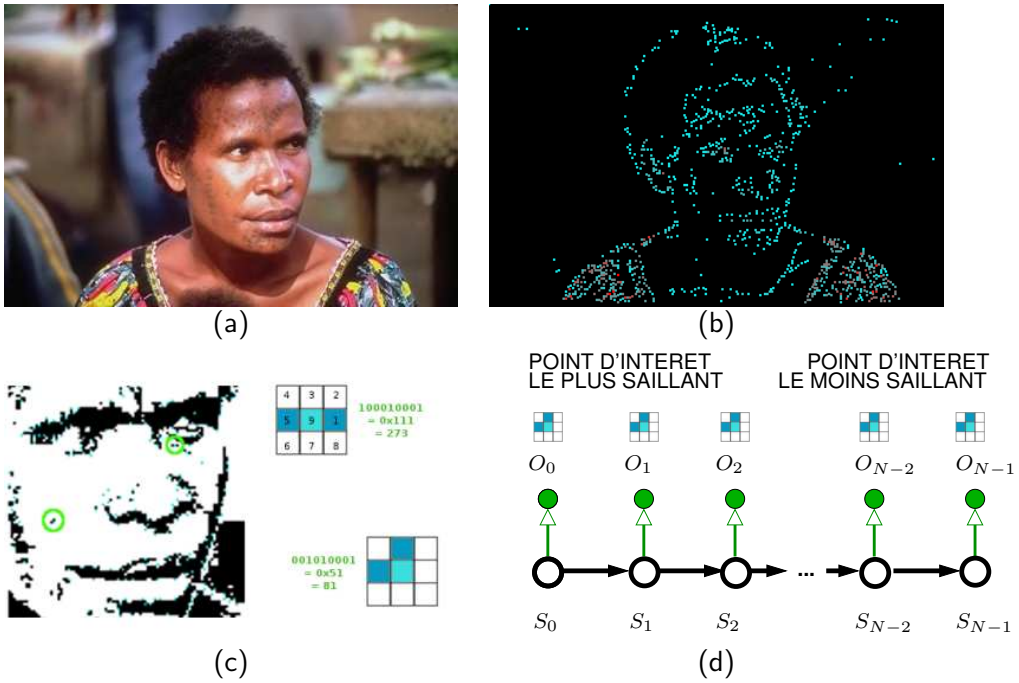


Figure 4.8 – Classification d'images par HMM : (a) image d'origine ; (b) points d'intérêt ; (c) un symbole par point ; (d) construction d'un HMM sur la séquence de symboles.

HMM est défini sur cette chaîne, où les observations du modèle correspondent aux caractéristiques extraites sur chaque point de la chaîne. Dans notre cas, une observation est un symbole discret correspondant aux valeurs binaires du voisinage 3×3 autour du point obtenues après une binarisation de l'image.

Durant la phase d'apprentissage, un modèle est appris pour chaque classe de la base à l'aide de l'algorithme Baum-Welch [Rab89]. Durant la phase de test, la probabilité *a posteriori* maximale va déterminer la classe prédite.

Nous avons effectuées des expériences sur la base SIMPLICITY [WLW01] qui comprend plusieurs classes ayant une description très « sémantique » : {*Peuples d'Afrique, plages, bâtiments, bus, dinosaures, éléphants, fleurs, nourriture, chevaux*}. Les résultats sont donnés dans la figure 4.3. L'intégration d'un couche cachée dans la modélisation améliore les résultats par rapport aux méthodes basées sur les histogrammes introduites dans [SJ07]. Les deux distances utilisées sont d_H, une distance d'histogramme classique, et d_{H_ω}, une distance où la contribution d'un point à un histogramme est pondérée par sa position dans la chaîne.

4.4 L'appariement de graphes et l'appariement par graphes

Dans le domaine de la vision par ordinateur, un grand nombre d'applications nécessitent, ou peuvent se résoudre par, l'appariement de deux ensembles de points. Dans ce mémoire cela concerne la détection et la reconnaissance d'objets et la détection et la reconnaissance d'actions. D'autres applications sont la reconstruction 3D et en général tous les problèmes faisant intervenir plusieurs vues. Les algorithmes de type RANSAC [FB81] se sont établis comme un quasi standard pour le

	500	*1000*	*2000*	*4000*
HMM(1)	63.1	63.2	62.7	62.8
HMM(2)	63.5	64.4	**68.1**	67.3
HMM(5)	63.4	64.9	67.1	70.0
HMM(10)	**64.5**	65.2	67.1	70.2
HMM(20)	62.6	65.0	66.6	70.1
HMM(50)	58.4	63.9	67.2	**70.6**
HMM(100)	51.8	60.4	66.1	70.3
KPPV(d_H)	63.2	64.3	64.0	62.8
KPPV(d_{H_ω})	63.0	66.3	67.6	66.2
GM(d_H)	63.4	65.9	58.0	50.8
GM(d_{H_ω})	61.6	**66.4**	65.9	60.8

Table 4.3 – Classification d'images par HMM : taux de classification pour 500, 1000, 2000 et 4000 points. HMM(T) signifie un HMM avec T états cachés. KPPV(d) signifie le k plus proche voisin par rapport à une distance d. GM(d) signifie la médiane généralisée [dlHC00] par rapport à la distance d.

cas où les « objets » (dans un sens large) à apparier sont rigides. Ils estiment un modèle globale et rigide (isométrie, transformé de similarité, homographie etc.) permettant de passer d'un ensemble de points à l'autre, en gérant de manière robuste les points aberrants.

Ce genre de méthodes échouera si les objets ne sont pas rigides, comme cela est le cas pour les humains et les animaux. Dans ces conditions, il faudrait restreindre la vérification d'invariances géométriques aux sous ensembles de points qui ne sont pas séparés par une articulation. En pratique cela s'avère difficile. L'appariement de graphes est une solution possible à ce problème. Pour cela, les deux nuages des points sont structurés en graphes en ajoutant des arêtes par des moyens divers, comme la triangulation ou par des critères de proximité ou d'adjacence. Notons qu'il est tout à fait possible de construire des graphes sur d'autres primitives que des points, comme par exemple des régions après une segmentation ou sur des morceaux de contours.

Ici nous reprenons de manière plus formelle le thème d'appariement non-rigide que nous avons touché, de manière plutôt intuitive, dans l'introduction de ce mémoire — voir la section 2.1 du chapitre 2, où nous nous sommes servis d'un raisonnement de type appariement de graphe afin de trouver un dessin modèle dans un dessin de scène.

De manière plus formelle, un modèle visuel doit être apparié avec une scène, la dernière typiquement étant plus grande, c.à.d. composée de plus de primitives. Chacun des deux ensembles est organisé comme un graphe $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, où $\mathcal{V}$ est l'ensemble de sommets, correspondant aux points, et $\mathcal{E} = \mathcal{V} \times \mathcal{V}$ est l'ensemble des arêtes. Notre notation distinguera entre le graphe du modèle $\mathcal{G}^{\langle m \rangle} = \{\mathcal{V}^{\langle m \rangle}, \mathcal{E}^{\langle m \rangle}\}$ composé de M sommets et le graphe de la scène $\mathcal{G}^{\langle s \rangle} = \{\mathcal{V}^{\langle s \rangle}, \mathcal{E}^{\langle s \rangle}\}$ composé de S sommets.

Un appariement entre deux graphes est une relation $a \subseteq \mathcal{V}^{\langle m \rangle} \times \mathcal{V}^{\langle s \rangle}$ associant des sommets du modèle aux sommets de la scène. Selon le problème étudié, l'appariement peut être bijectif ou injectif ; il peut être univoque ou multivoque, selon qu'il autorise ou non un sommet de $\mathcal{V}^{\langle m \rangle}$ (resp. $\mathcal{V}^{\langle s \rangle}$) à être apparié à exactement 1, au plus 1, ou un nombre quelconque de sommets de l'autre

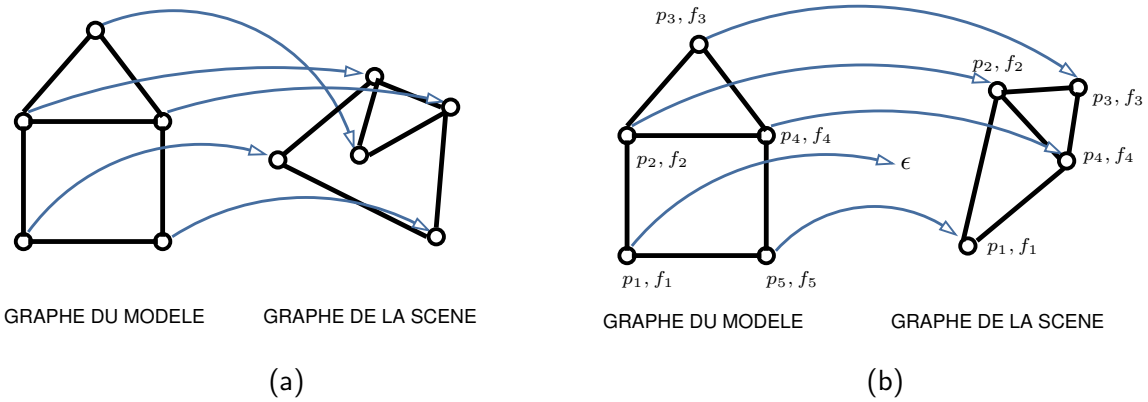


Figure 4.9 – Deux formalismes différents d'appariement de graphes : (a) l'appariement exact ; (b) l'appariement inexact tenant compte de la géométrie. Le sommet 1 du graphe de modèle n'est pas apparié (association au sommet fictif ϵ).

graphe.

L'appariement de graphes est un sujet traité parallèlement par les chercheurs d'au moins deux communautés : la communauté informatique théorique et théorie de graphes, et la communauté d'analyse d'images et de vidéos. Les formalismes étudiés sont légèrement différents, la différence étant le degré de prise en compte de la géométrie dans l'appariement. Cette distinction n'est bien sûr pas exacte, les deux communautés ne sont heureusement pas cloîtrées. Dans les grandes lignes, deux formulations se distinguent en deux groupes plus un groupe de méthodes plutôt numériques [4] :

Appariement exact Traité principalement dans la communauté de théorie de graphes, cette problématique demande un appariement stricte entre les deux graphes, appelé isomorphisme de graphes, ou entre le premier graphe et un sous-graphe du second graphe, appelé isomorphisme de sous-graphes. Ces méthodes sont aussi souvent qualifiées de méthodes « structurelles », du fait que l'appariement des sommets des graphes doit respecter la structure des deux graphes : si une arête existe entre deux sommets, une arête doit également exister entre les deux sommets appariés. Selon le formalisme, ce raisonnement peut également s'appliquer à l'absence d'arêtes (isomorphisme) ou pas (monomorphisme). La plupart des méthodes exactes ne recourent à aucune information supplémentaire à la structure des graphes, mais cela n'est pas une nécessité absolue. Un problème alternatif concerne la recherche du plus grand sous-graphe commun et donc de l'isomorphe de taille maximale des deux (sous)-graphes.

La figure 4.9a montre un exemple d'appariement exact entre deux graphes. Notons que les deux graphes appariés sont isomorphes, malgré la différence en géométrie. Dans le cas de deux graphes planaires, les deux graphes plans, c.à.d. les plongements des graphes dans le plan 2D, ne sont pas identiques : les faces du premier graphe ne correspondent pas aux faces du deuxième graphe.

Un autre inconvénient majeur de ces méthodes réside dans leur manque de robustesse face au bruit, comme la suppression ou l'insertion de sommets, ou à la modification de la structure due à un changement des positions des points.

4. Les noms donnés à ces formalismes peuvent différer, le terme même « exact » étant utilisé pour plusieurs notions aux seins des deux communautés

Appariement inexact Comme le nom l'indique, un appariement inexact de graphes va chercher un appariement des sommets sans que la structure des graphes soit nécessairement complètement identique, c.à.d. qu'un isomorphisme n'est pas demandé. Pour cela, une fonction est minimisée, nommée « fonction d'énergie » dans la communauté vision et « fonction objectif » dans la communauté recherche opérationnelle. La plupart des méthodes inexactes s'appuient sur des informations supplémentaires à la structure du graphe, des attributs habituellement associés à la géométrie de l'image et de la vidéo. Cela n'est pas un nécessité absolue, des méthodes inexactes sur des graphes sans attributs existent.

La figure 4.9a montre un exemple d'appariement inexact. A chaque sommet i d'un graphe est aussi associé une position dans l'image ou dans la vidéo, dénotée p_i, et un descripteur d'apparence locale f_i. Pour un appariement donné, des invariances géométriques sont vérifiées entre les points connectés par une arête.

Méthodes sans appariement explicite Par souci d'exhaustivité il convient de mentionner les méthodes permettant de comparer des graphes sans les apparier explicitement. Les méthodes dites « spectrales » obtiennent une description de la structure d'un graphe par les décomposition en vecteur propres d'une matrice décrivant la structure du graphe telle que la matrice d'adjacence ou la matrice Laplacienne. Cela permet d'en extraire des caractéristiques, d'explorer l'espace des changements de structure, de faire du *clustering* etc. — cf. [LWH03], entre autres. Certaines méthodes procèdent par plongement dans des espaces vectoriels, du graphe ou des sommets du graphe [XHW09]. Ce type de méthodes ne sera pas approfondi ici, nous renvoyons le lecteur à [HW12] pour un survey très récent. Notons que ces méthodes dites « spectrales » sont différentes de l'appariement « spectrale », exploré plus loin dans la section 4.4.2.

4.4.1 Appariement exact

Les applications du problème de l'isomorphisme de graphes à la reconnaissance d'objets sont peu fréquentes, car dans ce formalisme les deux graphes sont obligatoirement de la même taille. Il est intéressant à savoir que la classe de complexité de ce problème est une questions ouverte. Le problème fait clairement partie de la classe NP. Or, aucune preuve de NP-complétude est connue à ce jour. On conjecture que la complexité de ce problème se trouve entre les classes P et NP-complet (si P$\neq NP$)[5].

D'autre part, il est largement connu que l'isomorphisme de sous-graphes est NP-complet [GJ79]. La plupart des algorithmes connus sont basés sur une recherche dans un arbre codant les possibilités d'appariement en intégrant des mécanismes de retour en arrière (*backtracking*) en cas de blocage. Un exemple typique est l'algorithme de Ullmann [Ull76]. La méthode évite de traiter toutes les combinaisons possibles d'affectations en vérifiant certaines cohérences locales.

Les deux problèmes peuvent aussi être interprétés comme un problème de satisfaction de contraintes (CSP) [Reg95, Sol10] et les méthodes classiques de ce domaine peuvent être appliquées. On associe une variable à chaque sommet du modèle, le domaine de cette variable contenant l'ensemble des sommets auxquels il peut être apparié, c.à.d. l'ensemble des sommets de la scène. Les contraintes imposent que si deux sommets du modèle sont liés par une arête, les deux sommets affectés doivent être liés également (et vice-versa). La méthode est très similaire aux méthodes

5. Nous devons à Cagatay Dikici que $P = NP$ de manière triviale si N=1 . . .

mentionnées auparavant : on explore l'ensemble des appariements en construisant un arbre de recherche, en supprimant des possibilités grâce à des méthodes de filtrage. A titre d'exemples on peut citer les conditions suivantes :

— Un sommet de scène affecté à un sommet de modèle peut être supprimé des domaines des autres sommets du modèle, c.à.d. des listes de sommets cibles des sommets modèles.
— Les degrés des sommets affectés doivent être égaux pour un isomorphisme. Pour un monomorphisme, le degré d'un sommet de scène doit être supérieur ou égal au degré du sommet modèle auquel il a été affecté.
— Si un sommet de modèle a été affecté à un sommet de scène, toutes les arêtes avec les voisins déjà affectés doivent rester préservées par l'affectation.

D'autres filtres sont possibles et connus, dont certains sont un peu plus globaux. Le concept de cohérence locale interdit toute affectation de sommet, si cela réduit l'ensemble de possibilités pour un autre sommet du graphe à l'ensemble vide. Cela réduit de manière considérable la largeur de l'arbre, tout en augmentant la complexité de calcul du filtre.

Pour l'isomorphisme de graphes des approches alternatives existent, qui ne s'appuient pas sur une structure arbrique. L'algorithme *Nauty* [McK81], bien connu, est un des algorithmes les plus rapides pour la détection d'isomorphismes de graphes [FSV01]. Il introduit une forme canonique pour les graphes et procède en comparant ces formes. Dans [BM97, MB99], une bibliothèque de graphes connus est créée lors d'une phase hors-ligne, ensemble avec un arbre de décision par graphe. Ce pré-traitement permet d'apparier un nouveau graphe à cette bibliothèque avec une complexité de $O(N^2)$ par rapport à la taille du graphe.

4.4.2 Appariement inexact

Contrairement aux méthodes dites exactes, les méthodes inexactes permettent un appariement entre les deux graphes qui ne respecte pas à cent pour cent la structure des graphes, c.à.d. qu'il ne s'agit ni d'un isomorphisme ni d'un isomorphisme de sous-graphes. L'appariement s'appuie sur des attributs supplémentaires des graphes, extraits à partir de la géométrie de l'image et des informations d'apparence locale. Cela peut se formaliser de manière suivante.

A chaque sommet i des deux graphes (modèle et scène) est également associée une position spatio-temporelle $p_i = [\ p_i^{<x>}\ p_i^{<y>}\ p_i^{\langle t \rangle}\ ]^T$ et un vecteur de caractéristiques d'apparence f_i. Lorsque nécessaire, nous ferons une distinction entre le modèle et la scène par les exposants $^{\langle m \rangle}$ et $^{\langle s \rangle}$: $p_i^{\langle m \rangle}, f_i^{\langle m \rangle}, p_i^{\langle s \rangle}, f_i^{\langle s \rangle}$ etc. Notons que les symboles dans les exposants entourés de chevrons $\langle . \rangle$ ne sont pas des indices numériques ; il s'agit de symboles indiquant une catégorie. Dans le symbole $f_i^{\langle m \rangle}$, i peut prendre des valeurs dans $\{1, \ldots, M\}$; $\langle m \rangle$ ne pourra prendre des valeurs, il indique l'appartenance au modèle.

A chaque sommet i du graphe de modèle est associé une variable discrète $x_i,\ i = 1..M$, dont la valeur représente l'appariement du sommet à un sommet de la scène. Chaque x_i peut donc prendre des valeurs de l'ensemble $\{1 \ldots S, \epsilon\}$, où S est le nombre de sommets de la scène. La valeur $x_i = \epsilon$ signifie que le sommet i du modèle n'est pas apparié, une possibilité admis pour gérer les occultations. La valeur $x_i = j$ signifie que le sommet i du modèle est apparié au sommet j de la scène. L'ensemble complet de variables $x_i, i \in 1 \ldots M$ est aussi noté x.

En fonction du type d'appariement souhaité, une fonction globale, nommée « fonction d'énergie », est conçue mesurant la qualité d'un appariement. La solution est obtenue en minimisant cette fonction. A chaque combinaison d'affectations x correspond une valeur d'énergie donnée.

En principe, l'énergie sera petite pour des appariements qui correspondent à une transformation réaliste du modèle à la scène. Déterminer l'appariement optimal revient à minimiser la fonction d'énergie. Traditionnellement, les fonctions impliquent plusieurs types d'informations :

- des différences structurelles entre le graphe modèle et le graphe de scène : quelle est la longueur du chemin le plus court entre les deux sommets du graphe de scène dont les sommets appariés dans le graphe de modèle sont voisins ?
- des différences entre les caractéristiques d'apparence d'un sommet du graphe du modèle et les caractéristiques d'apparence du sommet de la scène associé ;
- des différences géométriques liées à l'appariement, par exemple le dégrée de préservation des distances ou des angles.

L'appariement inexact est souvent formalisé par une de trois manières fréquentes :

- La distance d'édition entre graphes a été développée au sein de la communauté informatique théorique et théorie de graphes.
- Les fonctions d'énergie existent sous deux formes principales : la forme de type *sommation* et la forme de type *matricielle*. Ces deux formalisations sont très répondues dans la communauté de vision par ordinateur.

La distance d'édition

La distance d'édition entre deux graphes se base sur un ensemble d'opérations élémentaires, dont la définition peut varier. Les opérations les plus fréquentes sont

- L'insertion d'un sommet
- La suppression d'un sommet
- L'insertion d'une arête
- La suppression d'une arête

Un coût est associé à chaque opération. La fonction objectif à minimiser correspond à la somme des coûts d'une séquence d'opération, dite « séquence d'édition ».

La forme de type sommation

Sous cette forme, la fonction d'énergie se décompose en sommes pondérées de termes unaires et binaires :

$$E(x) = \lambda_1 \sum_{i \in \mathcal{V}^{\langle m \rangle}} U(x_i) + \lambda_2 \sum_{(i,j) \in \mathcal{E}^{\langle m \rangle}} D(x_i, x_j) \tag{4.2}$$

Ici nous avons omis de la notation tous les arguments sur lesquels l'énergie ne sera pas minimisée. Il va sans dire que les attributs du graphe, donc les positions p_i et les caractéristiques d'apparence f_i, interviendront dans les termes $U(.)$ et $D(.,.)$.

Notons que avec ce genre de fonction d'énergie, on ne peut pas interdire que deux sommets du modèle soient appariés à un même sommet de la scène.

Les termes de type $U(.)$ modélisent le coût d'associer le sommet i du graphe du modèle au sommet x_i du graphe de scène. Typiquement ce coût est réalisé par une distance sur les caractéristiques $f_i^{\langle m \rangle}$ et $f_{x_i}^{\langle s \rangle}$. Les termes $D(.,.)$ vérifient des contraintes géométriques, typiquement la similarité des longueurs d'une arête et l'arête appariée. Ces contraintes ne sont pas invariantes à l'échelle, ce qui rend leur application difficile. Ce problème peut être résolu en échangeant une modélisation par graphe par une modélisation par hyper-graphes. Rappelons qu'un hyper-graphe

est une généralisation d'un graphe, où une hyper-arête peut connecter n'importe quel nombre de sommets, généralement plus de deux [ZS08]. Cela rend possible la vérification des contraintes exprimées sur des triplets de sommets, par exemple en se basent sur des angles [LH05]. La fonction d'énergie se décompose alors en termes unaires et ternaires :

$$E(x) = \lambda_1 \sum_{i \in \mathcal{V}^{\langle m \rangle}} U(x_i) + \lambda_2 \sum_{(i,j,k) \in \mathcal{E}^{\langle m \rangle}} D(x_i, x_j, x_k) \tag{4.3}$$

Dans ce cas, les hyper-arêtes dans $\mathcal{E}$ connectent des ensembles de trois sommets, donc des triangles. A partir de maintenant nous allons souvent et abusivement appeler, le cas échéant, « graphes » les hyper-graphes et « arêtes », ou « triangles, » les hyper-arêtes.

Il est intéressant de noter que la forme de type sommation indique la possibilité d'une interprétation Markovienne de cette modélisation. En effet, la fonction d'énergie de type 4.2 peut être interprétée comme l'énergie associée à un MRF — voir aussi la section 3.1.4 du chapitre 3. En posant un cadre Bayesien, on peut interpréter les termes binaires comme l'information *a priori* du modèle et les termes unaires comme les termes d'attache aux données. Il est donc peu étonnant que certaines méthodes d'appariement se servent d'algorithmes de minimisation d'énergie issus des modèles graphiques probabilistes comme l'algorithme *junction tree* utilisé dans [TCSB06] ou des méthodes de la famille *graph cuts* utilisées dans [TKR08, ZWW+10].

La forme matricielle

Sous cette forme alternative, une solution donnée est modélisée comme une matrice $X_{i,j}$ de taille $M \times S$ de valeurs binaires, tel que $X_{i,j} = 1$ si le sommet i du modèle est apparié au sommet j de la scène. La contrainte suivante est classiquement imposée :

$$\sum_i X_{i,j} = 1 \tag{4.4}$$

Cette matrice joue donc le rôle de l'ensemble des variables x_i introduit auparavant, les mêmes valeurs sont admissibles.

Il est commode de vectoriser, ligne par ligne, la matrice X en un vecteur x de dimension MS — à ne pas confondre avec l'ensemble x dans la forme de type sommation. Dans cette représentation, une affectation possible d'un sommet i vers un point j correspond à l'indice $iS+j$ dans le vecteur, que nous dénoterons comme $\mapsto_{ij} = iS+j$ dans la suite. Cela permet d'écrire la fonction d'énergie comme une forme quadratique :

$$E(x) = x^T A x \tag{4.5}$$

La contrainte (4.4) est donc reformulé comme suit : $\sum_i x_{\mapsto_{ij}} = 1$. Les termes unaires et binaires de la fonction d'énergie ont été intégrés dans la matrice A :

- les potentiels unaires interviennent dans la diagonale de la matrice A, c.à.d. le coût de l'affectation du sommet i au sommet j correspond à la valeur $A_{\mapsto_{ij},\mapsto_{ij}}$;
- les potentiels binaires interviennent dans les valeurs hors diagonale de la matrice A. Le coût de l'affectation de l'arête (i,k) du modèle à l'arête (j,l) de la scène correspond à la valeur $A_{\mapsto_{ij},\mapsto_{kl}}$.

L'extension aux hyper-graphes d'ordre supérieur est possible [DJP11].

Minimisation

Les deux formalisations décrites ci-dessus sont fréquemment utilisées dans la littérature. Rappelons qu'elles sont mathématiquement équivalentes. La première formulation, basée sur les sommes, se prête plus à des algorithmes de type optimisation discrète, tel que les *graph cuts*. Plusieurs méthodes décomposent le problème initial en sous-problèmes qui sont résolus avec les outils d'optimisation discrète comme des *graph cuts* [TKR08, ZWW+10]. Dans [DJP11], un algorithme d'optimisation de type *graph cuts* est étendu aux multi-labels et aux 2D en alternant entre les coordonnées x et y.

La deuxième formulation se prête à une solution plus orientée algèbre linéaire. Dans les méthodes dites « spectrales », par exemple [LH05], le domaine de x dans l'équation (4.5) est relâché permettant aux éléments du vecteur x de prendre des valeurs réelles dans $[0, 1]$ au lieu de valeurs binaires $\{0, 1\}$, tout en gardant des contraintes de normes. Il peut être démontré que le minimum de (4.5) est le vecteur propre principal de la matrice A. La solution de l'appariement est obtenue en seuillant les résulats réels pour obtenir la matrice binaire décrivant les affectations. Cela n'est pas optimal puisque la solution optimale binaire (du problème NP-complet d'origine) n'est pas forcement obtenue en passant par une solution du problème relâché réel. Dans [DBKP09], cette approche est étendue aux hyper-graphes d'ordre trois et le vecteur propre principal est calculé de manière efficace et approchée avec un algorithme itératif.

Dans [ZS08] une autre méthode est présentée qui procède par une relaxation du problème d'origine. L'appariement des sommets est traduit en un appariement des arêtes. Dans un cadre probabiliste, le problème est simplifié en supposant l'indépendance conditionnelle entre affectations d'arêtes. L'algorithme cherche la matrice d'affections de sommets la plus probable expliquant les affectations d'arêtes.

Dans [ZBV09], une approche de programmation convexe-concave est employée sur un problème de moindres carrés sur les matrices de permutation. Dans [LZLZ09], une structure de graphe candidat est créée et le problème est formulé comme un problème de coloration de graphes organisé en plusieurs couches. Une solution pour le problème résultant de programmation d'entier quadratique est proposée dans [LZS11] ; dans [LCL11], le problème est étendu aux relations d'ordre général (> 3) et résolu avec des marches aléatoires. Dans [TCSB06], la structure du graphe de modèle est approximée par un k-tree, permettant de résoudre le problème de manière exacte.

Dans les sections suivantes, nous présenterons nos travaux sur l'appariement de graphes en vision par ordinateur : la reconnaissance d'objets dans la section 4.5, et la reconnaissance d'actions dans la section 4.6.

4.5 La reconnaissance d'objets

Le travail présenté dans cette section a été réalisé dans le cadre d'un contrat de type CIFRE avec *Pinka Studios*, une entreprise spécialisée sur la création de films animés [TWLB09]. Le processus de création de films d'animation 3D commence avec le dessin manuel de storyboards à l'aide de crayons classiques ou électroniques. Les scènes 3D sont ensuite modélisées de manière manuelle à partir de ces storyboards dans un processus fastidieux, d'où la volonté d'exécuter cette partie automatiquement avec un algorithme de vision par ordinateur. L'objectif est de bénéficier de la réutilisation des modèles 3D entre les films. En effet, pour chaque épisode, les spécialistes de la

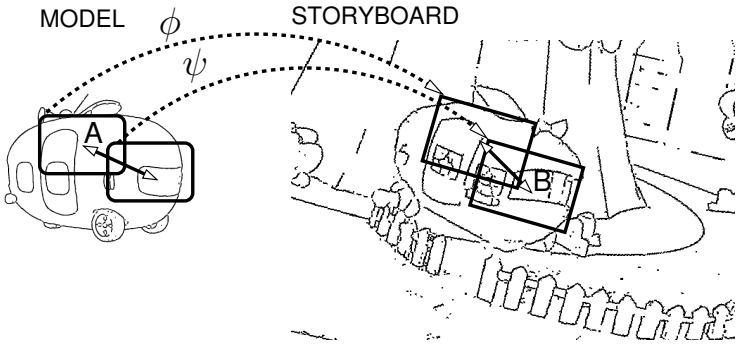


Figure 4.10 – Les différentes contraintes : I. La distance de Zernike z_d entre les patches du modèle et les patches de la scène. ; II. la distance Euclidienne A entre deux patches voisins doit être cohérente avec la distance Euclidienne B des patches appariés ; III. l'angle de rotation ϕ entre un sommet et le sommet apparié doit être cohérent avec un appariement voisin.

modélisation utilisent des modèles d'objets 3D existants piochés dans une base de données. L'objectif de l'algorithme proposé est de reconnaître chaque modèle 3D à partir d'un dessin, ensemble avec sa pose, c.à.d. les angles de rotations, afin de pouvoir le placer dans la scène 3D de manière automatique.

Le problème de connaissance 3D peut être traduit en un problème 2D en représentant chaque modèle 3D par un ensemble de vues, tant que le nombre de vues est suffisamment grand par rapport aux exigences sur la précision de la pose à estimer. Cela permet de traiter le problème de reconnaissance comme un problème d'appariement entre une image 2D du dessin et une image 2D de la scène, où chaque image de dessin correspond à une seule vue d'un objet 3D.

Inspiré du travail de Revaud et al. [RLB09], nous avons choisi des descripteurs basés sur les moments de Zernike, une décomposition d'une imagette très adaptée à la reconnaissance de formes binaires. Cependant, nous les avons employés de manière locale, ce qui nous permet de surmonter les problèmes d'occultation et de transformations non-rigides. Le problème a été formulé comme un problème d'appariement de graphes comme décrit dans la section 4.4.2. Le graphe d'un modèle est construit en le couvrant de patches (de petites imagettes) de manière dense. Le descripteur d'apparence f_i associé à chaque sommet i du graphe correspond à quelques moments Zernike de basses fréquences.

Une contribution consiste en l'introduction de l'angle de rotation entre deux patches dans le formalisme d'appariement. En effet, la distance de Zernike introduite dans [RLB09] permet, à partir de deux vecteurs de caractéristiques f_a et f_b, un calcul efficace de deux mesures :

— une distance entre les deux vecteurs, notée $z_d(f_a, f_b)$;
— l'angle de rotation entre les deux vecteurs, noté $z_a(f_a, f_b)$, supposant qu'il s'agit du même motif.

La distance z_d intervient naturellement dans les termes unaires $U(.)$ de l'équation (4.2) :

$$U(x_i) = z_d\left(f_i^{\langle m \rangle}, f_{x_i}^{\langle s \rangle}\right) \tag{4.6}$$

tandis que les termes binaires $D(.,.)$ sont basés sur une différence $D_l(.,.)$ sur les longueurs d'arêtes

	Globale [RLB09]		Par graphe	
	total	%	total	%
tents	4/8	50.00	7/8	**88.00**
trailers	2/3	**67.00**	2/3	**67.00**
bushes	3/10	30.00	6/10	**60.00**
trees	1/31	3.00	4/31	**13.00**

	Global [RLB09]	Par graphe
tents	0.81	**0.31**
trailers	0.29	**0.06**
bushes	**1.16**	1.41
trees	2.70	**1.36**

Table 4.4 – Reconnaissance d'objets dessinés. (a) rappel pour 100% de précision ; (b) erreur moyenne de détection de point de vue.

et une distance $D_a(.,.)$ entre angles, tenant compte de leur domaine circulaire :

$$\begin{aligned} D(x_i, x_j) &= D_l\left(||p_i^{\langle m\rangle} - p_j^{\langle m\rangle}||, ||p_{x_i}^{\langle s\rangle} - p_{x_j}^{\langle s\rangle}||\right) + \\ &+ D_a\left(z_a(f_i^{\langle m\rangle}, f_{x_i}^{\langle s\rangle}), z_a(f_j^{\langle m\rangle}, f_{x_j}^{\langle s\rangle})\right) \end{aligned} \tag{4.7}$$

Notons que les deux arguments de la distance $D_l(.,.)$ sont calculés, respectivement, sur une arête et son arête associée, tandis que les deux arguments de la distance $D_a(.,.)$ sont calculés, respectivement, sur un appariement de sommet et un appariement voisin. Cela est illustré dans la figure 4.10. Une autre particularité de l'approche est l'absence de termes sur la cohérence structurelle : les arêtes du « graphe » de scène n'interviennent pas dans la fonction d'énergie. Il n'y a donc aucune obligation de lier les sommets de la scène en un graphe. Dans le contexte applicatif, cela ne s'avérait pas nécessaire.

La reconnaissance nécessite la minimisation de l'équation (4.2) avec les termes $U(.)$ et $D(.,.)$ donnés par (4.6) et (4.7). Dans ce travail nous tenons compte de la nature des images et du fait que les descripteurs de Zernike sont assez puissants sur les images de type dessins. Au lieu de résoudre le problème d'origine, nous séparons la fonction d'énergie (4.2) en deux parties et nous procédons en deux étapes :

1. L'appariement est fait avec les termes unaires $U(.)$ seulement, c.à.d. à partir des distances de Zernike $z_d(.,.)$. Les termes binaires $D(.,.)$ sont ignorés. Le calcul revient à une recherche de distance minimale par sommet, un calcul qui peut être accéléré à l'aide de structures de données comme les arbres k-d.
2. La reconnaissance proprement dite est faite dans une deuxième étape, en vérifiant les termes binaires $D(.,.)$ sur les sommets appariés. Parmi tous les sommets du modèle, un sommet unique est sélectionné qui minimise la somme des termes binaires dans lesquels il intervient. Cette somme est également utilisée comme score de détection et de reconnaissance.

Cette manière simplifiée de calculer l'appariement a plusieurs mérites :

— la méthode est simple et efficace,
— les occultations sont gérées de manière évidente, puisque les sommets impossibles à apparier ne peuvent pas perturber les appariement voisins.

La figure 4.11 montre quelques exemples de détection et de reconnaissance sur des images de storyboard fournies par l'entreprise *Pinka*. Nous pouvons observer que les vues des modèles 3D sont détectées même lorsqu'elles sont occultées sévèrement.

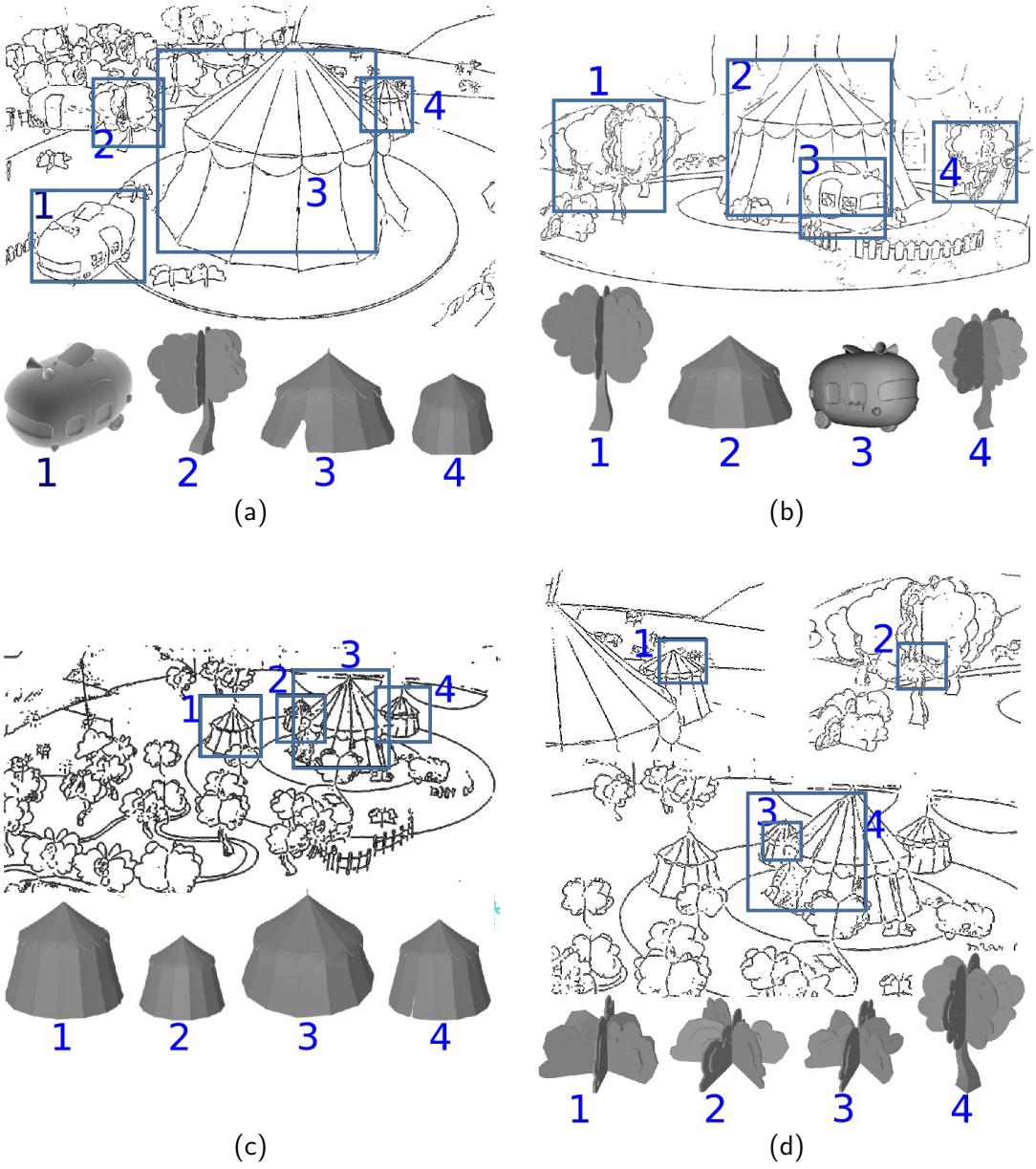


Figure 4.11 – Exemple de détection sur des storyboards de l'entreprise *Pinka*. La reconnaissance a été paramétrée pour une précision de 100% ; (d) Les modèles de type arbre et brousse posent quelques problèmes du aux grandes variations de type intra-classe.

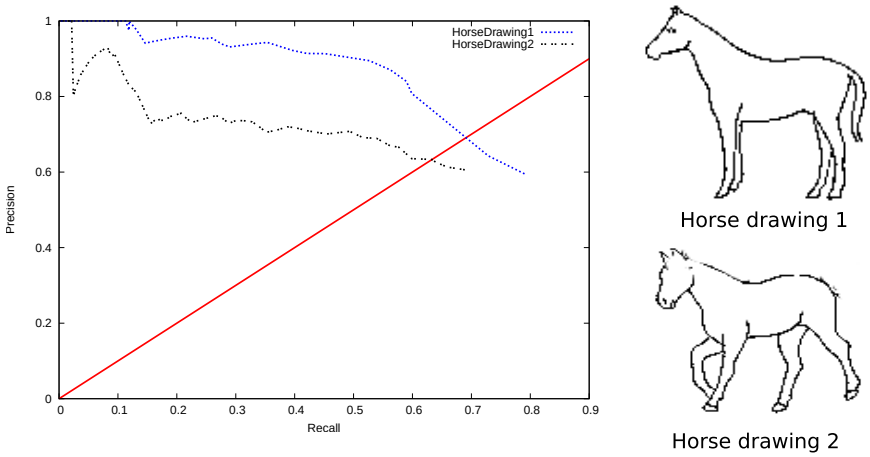


Figure 4.12 – Un exemple montrant la stabilité de la méthode face aux variations d'un objet non-rigide de la base *Weizmann-Shotton* [SBC05].

Comme la plupart des algorithmes de détection classiques, notre algorithme dépend d'un seuil agissant sur la fonction de score. Ce seuil permet traditionnellement de gérer le compromis entre rappel et précision de la détection, donc entre le nombre de faux négatifs et le nombre de faux positifs. Ici, l'algorithme a été réglé de manière à éviter des faux positifs, c.à.d. les résultats correspondent à la mesure de rappel pour une mesure de précision égale à 100%.

La table 4.4a montre une comparaison avec la méthode globale de Revaud et al. [RLB09], indiquant une bien meilleure performance de notre méthode. Dans la table 4.4b l'erreur d'estimation de point de vue est donnée. A chaque vue d'un modèle 3D sont associée deux angles de rotation, que nous traduisons en un point 3D sur une sphère unité. L'erreur donnée est la moyenne de la distance Euclidienne entre les points estimés et le point connu de la vérité de terrain.

La méthode a également été testée sur deux bases d'images connues, la base ETHZ proposée dans [FTG06] et la *Weizmann-Shotton* proposée dans [SBC05]. S'agissant d'images naturelles, et non pas d'images dessinés, le contexte applicatif est plus général que le contexte industriel initial. Afin de pouvoir appliquer notre méthode, un détecteur classique de contours a transformé les images naturelles en images ressemblant à des dessins. Les images de requête ont été de type dessin.

La figure 4.13 montre des résultats visuels sur la base ETHZ en cherchant les objets *pomme, cygne, mug, giraffe et bouteille*. Nous pouvons remarquer que les objets ont été trouvés même en présence de déformations non-rigides, comme cela est souvent le cas avec les « objets » articulés tel que les animaux et les humains. Nous avons également effectué une expérience spécifique pour mesurer la robustesse face à ce problème, dont les résultats sont illustrés dans la figure 4.12. Des requêtes d'objets de type « cheval » ont été effectuées à partir de deux dessins modèle de forme différente, présentés à droite de la figure 4.12. Les courbes de précision/rappel montrent une performance similaire.

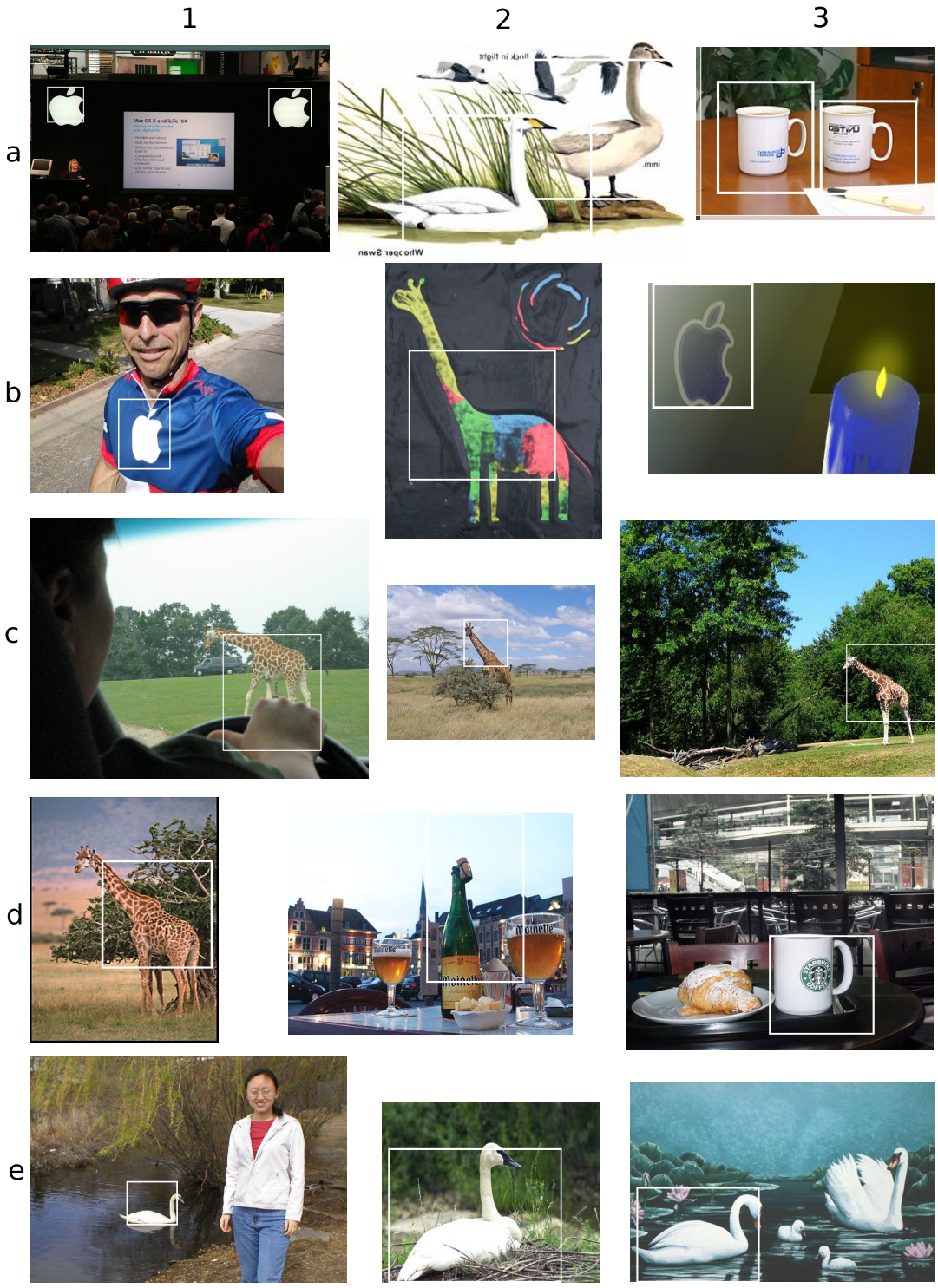


Figure 4.13 – Résultats sur la base *ETHZ* [FTG06].

4.6 La reconnaissance d'activités

Dans cette section nous présentons une autre application liée à l'analyse de scènes complexes : la reconnaissance d'activités dans des scènes avec des acteurs multiples ayant des comportements différents. Le problème est difficile, surtout dans un contexte de capteurs mobiles (téléphone portable, robot mobile) et donc dans un environnement de faible puissance de calcul. Les principaux verrous scientifiques concernent l'estimation et la caractérisation du mouvement humain ; la mise en correspondance sous contraintes de transformations non-rigides ; la gestion de l'invariance géométrique et de l'invariance par rapport au point de vue ; les contraintes de complexité ; l'intégration du contexte etc.

Faute de place, nous ne donnerons pas ici un état de l'art sur la reconnaissance d'activités dans la vidéo, un sujet très vaste. Une courte bibliographie restreinte aux modèles (semi)-structurées a été donnée dans la section 4.2. Nous la compléterons ici par quelques travaux sur l'appariement de graphes dans ce contexte.

Peu de travail existe sur l'appariement de graphes pour la reconnaissance d'activités. A notre connaissance, notre travail présenté dans la sous-section 4.6.2, publié dans [TWLB10], est la première publication basée sur un formalisme d'appariement de graphes dans un contexte de reconnaissance d'actions. Depuis, quelques autres travaux ont été publiés. Dans [BT11], des graphes construits à partir de sous-segmentations de vidéos sont appariés, également avec une méthode spectrale. Dans [GZSRC11], un modèle nommé *Strings of feature graphs* consiste à créer des chaines où chaque élément est un graphe construit à partir d'un petit segment d'une vidéo. Les vidéos sont appariées à l'aide d'un algorithme de type programmation dynamique (voir aussi la section 3.4.1 du chapitre 3) qui fait intervenir des micro-appariements pour les différents graphes des deux chaînes.

Par rapport aux autres modèles existants conçus pour la reconnaissance d'actions, illustrés dans la figure 4.2, l'appariement de graphes se situe à l'extrême droite : il s'agit d'une méthode tenant compte de toute la richesse d'une représentation structurelle. Les relations spatio-temporelles ne sont pas uniquement intégrées de manière statistiques ; elles sont vérifiées par un appariement point par point.

Dans le reste de ce chapitre nous traitons une modélisation de séquences vidéo par ensemble de points d'intérêts spatio-temporels, une extension des points d'intérêts spatiaux classiques à la dimension temporelle. Plusieurs détecteurs ont été proposés durant les dernières années, tels que les points périodiques [DRCB05] ; l'extension du détecteur Harris à la dimension temporelle [LL03] ; l'extension de SIFT [SAS07] ; le Hessien 3D [WTG08] et un détecteur basé sur les filtres de Gabor [NHH07]. Leur point commun est la tentative de sélectionner un ensemble parcimonieux de points dans une vidéo permettant de caractériser le mouvement. L'objectif est une invariance par rapport aux changements d'échelle, à la rotation etc., ainsi qu'une grande stabilité face aux perturbations classiques comme les bruits divers, le transcodage de la vidéo etc.

Ces détecteurs sont traditionnellement combinés avec des descripteurs calculés sur un patche spatio-temporel autour de chaque point : la concaténation de gradients [DRCB05] ; les *jets* spatio-temporels [SLC04] ; les histogrammes orientés de gradients et de flot optiques (HoG and HoF) et les descripteurs SIFT étendus à la dimension temporelle [SAS07].

Dans le reste de ce chapitre, trois contributions seront présentées :

— Dans la sous-section 4.6.2, une méthode de reconnaissance d'actions par appariement de

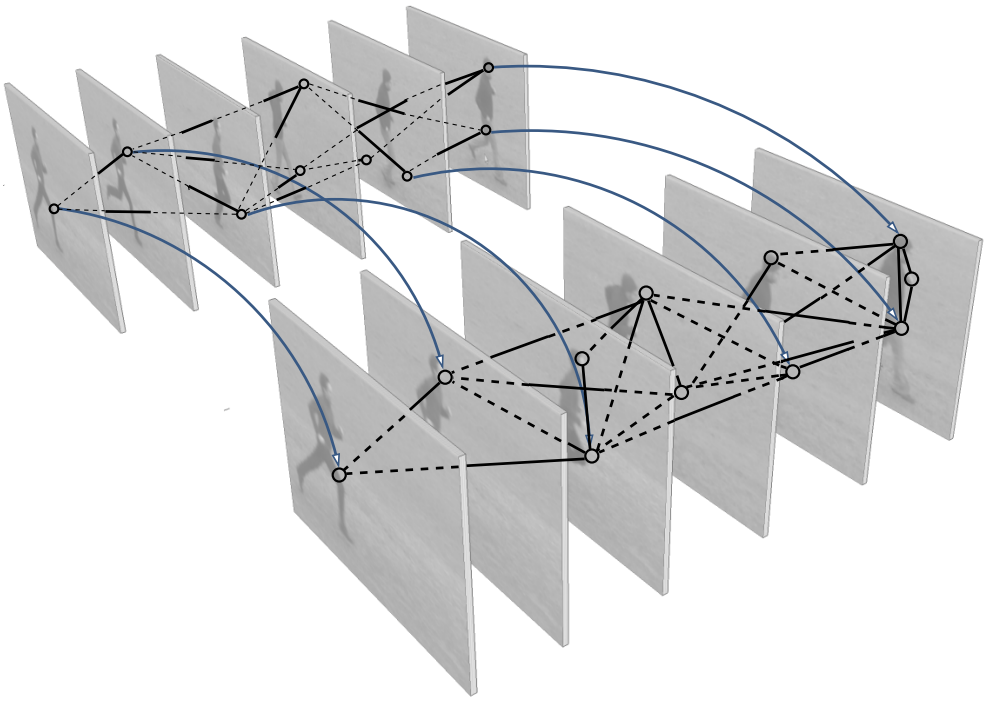


Figure 4.14 – Appariement de deux hyper-graphes construits à partir de séquences de vidéos.

hyper-graphes est présentée. Chacun des deux graphes à apparier est construit à partir de points d'intérêts spatio-temporels : un graphe pour chaque modèle d'action, et un graphe est construit pour chaque vidéo de scène. L'appariement est résolu de manière approchée avec un algorithme spectral classique.

— Dans la sous-section 4.6.3, une deuxième contribution traite le problème d'optimisation discrète qui est au cœur des algorithmes d'appariement. Nous nous focalisons sur le cas où les données sont plongées dans un « espace » spatio-temporel. Nous montrons que, dans ce contexte, nous pouvons profiter des propriétés particulières du domaine temporel, notamment la causalité et l'ordre strict imposé par cette dimension. Nous montrons que la complexité du problème est inférieure à la complexité de la problématique générale et nous dérivons un algorithme calculant la solution exacte.

— Dans la sous-section 4.6.4 nous profitons du résultat ci-dessus pour proposer une nouvelle méthode rapide de reconnaissance d'actions s'appuyant sur un graphe de structure spécifique, conçu pour un appariement rapide.

Le cas général, commun à tous les trois scénarios, est illustré dans la figure 4.14. Les M points de la vidéo modèle sont indexés arbitrairement avec des indices $i \in \{1, \ldots, M\}$ et les S points de la vidéo de scène également indexés arbitrairement avec des indices $i \in \{1, \ldots, S\}$. Les points sont structurés sous forme de hyper-graphes $\mathcal{G}^{\langle m \rangle} = \{\mathcal{V}^{\langle m \rangle}, \mathcal{E}^{\langle m \rangle}\}$ et $\mathcal{G}^{\langle s \rangle} = \{\mathcal{V}^{\langle s \rangle}, \mathcal{E}^{\langle s \rangle}\}$, respectivement. A chaque point i sont associés une position p_i et un vecteur de caractéristiques (un descripteur) f_i — voir également le début de la section 4.4.

Nous travaillons sur des hyper-graphes au lieu de graphes classiques afin de pouvoir introduire des termes géométriques sur les triplets de points — voir ci-dessous. Les hyper-arêtes de l'ensemble $\mathcal{E}$ de notre graphe connectent donc des ensembles de trois sommets. A partir de maintenant nous

allons abusivement appeler « graphes » les hyper-graphes et « arêtes », ou « triangles », les hyper-arêtes.

L'appariement se base sur une fonction d'énergie sous forme (4.3). La distance U est définie comme la distance Euclidienne entre deux vecteurs de caractéristiques, en tenant compte d'un paramètre de punition W^P si l'appariement n'a pas lieu :

$$U(x_i) = \begin{cases} W^p & \text{si } x_i = \epsilon \\ ||f_i^{\langle m \rangle} - f_{x_i}^{\langle s \rangle}|| & \text{sinon} \end{cases} \tag{4.8}$$

Comme nos données sont plongées dans l'espace-temps, la distorsion géométrique est traitée de manière séparée pour la partie spatiale et pour la partie temporelle :

$$D(x_i, x_j, x_k) = D^t(x_i, x_j, x_k) + \lambda_3 D^g(x_i, x_j, x_k) \tag{4.9}$$

où la distorsion temporelle D^t est définie comme une différence tronquée entre deux paires de sommets d'un triangle :

$$D^t(x_i, x_j, x_k) = \begin{cases} W^t & \text{if } \Delta(i,j) > T^t \ \vee \Delta(j,k) > T^t \\ \Delta(i,j) + \Delta(j,k) & \text{else} \end{cases} \tag{4.10}$$

Ici, T^t est un seuil à partir duquel un distance est considérée comme « non-validée », et $\Delta(a,b)$ est la distorsion temporelle due à l'appariement de la paire de sommets (a,b) :

$$\Delta(a,b) = |(p_a^{\langle m \rangle \langle t \rangle} - p_b^{\langle m \rangle \langle t \rangle}) - (p_{x_a}^{\langle s \rangle \langle t \rangle} - p_{x_b}^{\langle s \rangle \langle t \rangle})| \tag{4.11}$$

La distorsion spatiale D^g est définie sur des différences d'angles :

$$D^g(x_i, x_j, x_k) = \left|\left| \begin{array}{c} \phi^{\langle m \rangle}(i,j,k) - \phi^{\langle s \rangle}(x_i, x_j, x_k) \\ \phi^{\langle m \rangle}(j,i,k) - \phi^{\langle s \rangle}(x_j, x_i, x_k) \end{array} \right|\right| \tag{4.12}$$

Ici, $\phi^{\langle m \rangle}(a,b,c)$ et $\phi^{\langle s \rangle}(a,b,c)$ sont des angles sur le point b pour, respectivement, les triangles du modèle de la scène indexés par (a,b,c).

4.6.1 Les propriétés des données spatio-temporelles

Les données issues de vidéos sont caractérisées par des propriétés spécifiques à elles, surtout dues aux particularités de la dimension temporelle. Nous avons identifié trois hypothèses qui seront exploitées dans les deux méthodes présentées dans les sous-sections 4.6.2 et 4.6.3 :

Hypothesis 1 : Causalité — Les dimensions spatiales (x, y) et temporelles (t) ne doivent pas être traitées de la même manière. Alors que les objets et les humains peuvent subir des transformations géométriques arbitraires comme la rotation, les actions humaines ne peuvent normalement pas être inversées. Dans un appariement correct, l'ordre temporel des points doit donc rester intact, ce qui peut être formalisé comme suit :

$$\forall\ i,j : p_i^{\langle m \rangle \langle t \rangle} \leq p_j^{\langle m \rangle \langle t \rangle} \Leftrightarrow p_{x_i}^{\langle s \rangle \langle t \rangle} \leq p_{x_j}^{\langle s \rangle \langle t \rangle} \tag{4.13}$$

Rappelons nous que l'exposant $\langle t \rangle$ sélectionne la dimension temporelle ; il ne s'agit pas d'un indice.

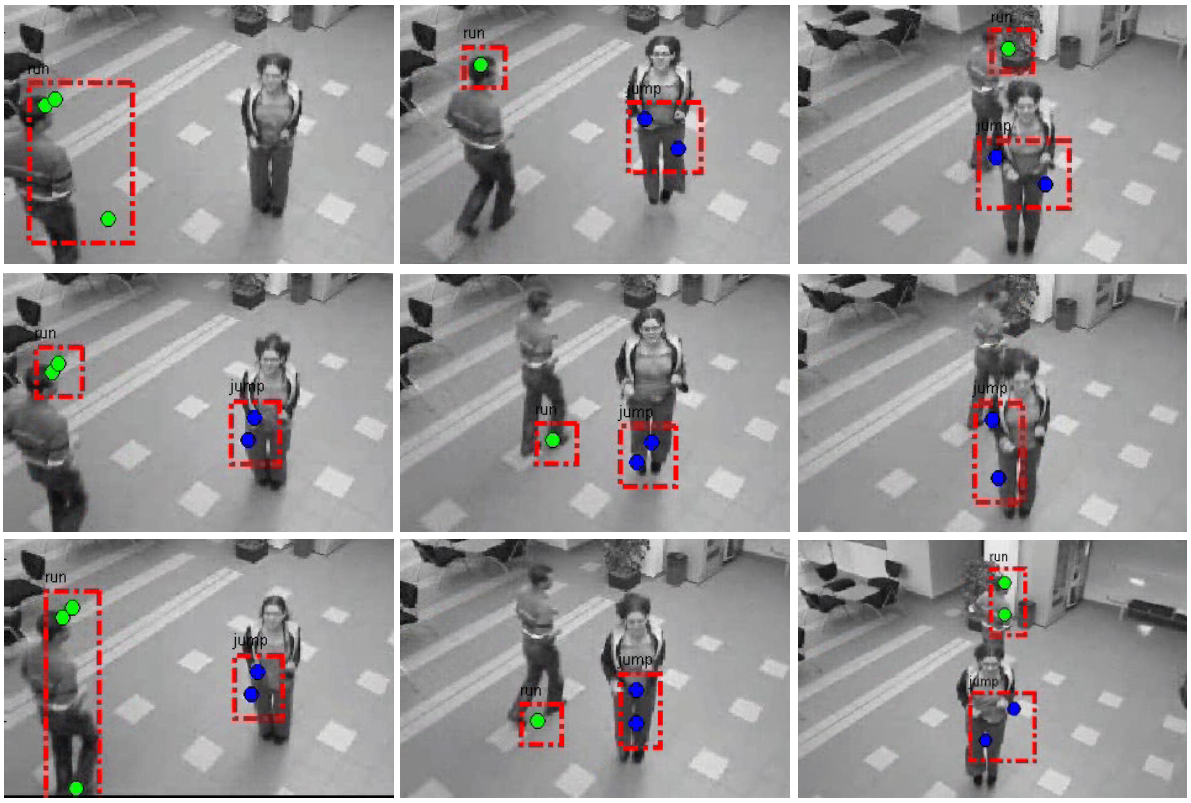


Figure 4.15 – Appariement approché de graphes : quelques résultats visuels.

Hypothesis 2 : Proximité temporelle — Une autre hypothèse raisonnable limite la quantité de distorsion temporelle entre les deux séquences. En d'autres termes, deux points proches dans le temps dans le modèle doivent être appariés à deux points proches dans le temps dans la scène. En supposant que le graphe du modèle est construit à partir d'informations de proximité, c.à.d. en seuillant les distances entre les points d'intérêt spatio-temporels, ce critère peut être formalisé comme suit :

$$\forall\ i,j,k \in \mathcal{E} : |p_{x_i}^{\langle s\rangle\langle t\rangle} - p_{x_j}^{\langle s\rangle\langle t\rangle}| < T^t \ \vee \ |p_{x_j}^{\langle s\rangle\langle t\rangle} - p_{x_k}^{\langle s\rangle\langle t\rangle}| < T^t \tag{4.14}$$

Hypothesis 3 : Unicité des instants temporels — Nous supposons qu'un instant ne peut être divisé ou fusionné avec un autre instant. En d'autres termes, tous les points d'une frame unique du modèle doivent être appariés avec les points d'une seul frame unique également :

$$\forall i,j : (p_i^{\langle m\rangle\langle t\rangle} = p_j^{\langle m\rangle\langle t\rangle}) \Leftrightarrow (p_{x_i}^{\langle s\rangle\langle t\rangle} = p_{x_j}^{\langle s\rangle\langle t\rangle}) \tag{4.15}$$

Dans la suite, ces hypothèses seront traduites en contraintes supplémentaires, ce qui nous permettra de rendre l'appariement plus efficace.

4.6.2 Appariement approché

Les méthodes classiques de type spectral s'appuient sur la forme matricelle de la fonction d'énergie (4.5), par exemple [LH05] et [DBKP09] — voir aussi la section 4.4.2. Elles procèdent en résolvant

un problème de valeur propre sur la matrice A décrivant les contraintes du système. Nous avons proposé une méthode de reconnaissance d'activités basée sur l'appariement de graphes basée sur une méthode spéctrale, tout en intégrant plusieurs améliorations. A notre connaissance, il s'agit de la première fois qu'un algorithme d'appariement de graphes a été utilisé sur des données spatio-temporelles [TWLB10]. Plusieurs autres travaux ont été publiés depuis [BT11, GZSRC11].

Formulation du problème — deux graphes sont appariés, un graphe pour le modèle et un graphe pour la scène. Les sommets sont liés aux points d'intérêts spatio-temporels extraits avec un détecteur classique tel que celui de Dollar [DRCB05]. La structure est déterminée en seuillant les distances entres les sommets. La fonction d'énergie suit la forme (4.5) avec les potentiels écrits par les équations (4.8) à (4.12).

Contraintes structurelles — afin de tenir compte des informations structurelles des deux graphes, les termes $D()$ sont mis à zéro pour les triangles du modèle dont l'affectation ne correspond pas à un triangle dans la scène. Il s'agit donc bien d'un algorithme d'appariement de graphes et non pas d'un algorithme de appariement de nuages de points par graphe.

Causalité — l'hypothèse nr.1 (« causalité ») décrite dans la sous-section 4.6.1 est exploitée pour rendre plus creuse la matrice de contraintes. Cela a deux effets : (i) l'algorithme d'appariement est plus efficace ; (ii) il produit des solutions de meilleure qualité grâce à la suppression des appariements contenant une inversion de l'ordre temporel des sommets.

Contrairement à la plupart des modèles semi-structurés, ce framework permet une classification et une localisation jointe des activités. L'appariement des sommets peut servir à localiser l'action parmi plusieurs actions dans la scène, comme il est illustré dans la figure 4.15 .

4.6.3 Appariement exacte

La méthode décrite dans la sous-section 4.6.2 s'appuie sur un algorithme de minimisation classique de type spectrale [DBKP09], qui calcule en temps polynomiale un bon appariement, sans aucune garantie d'optimalité. Dans cette sous-section nous introduisons une nouvelle méthode de minimisation exacte, c.à.d. le calcul du minimum exact et global de l'équation (4.3), avec des potentiels définis dans les équations (4.8) à (4.12) [CWS12b]. Dans le cas général, ce problème est connu comme étant NP-difficile [TKR08]. Par contre, nous allons montrer que le problème est plus facile si les données attribuées aux graphes sont plongées dans un espace contenant une dimension temporelle, c.à.d. si elles satisfont les contraintes dues aux hypothèses nr. 1 à 3 de la sous-section 4.6.1.

Notre méthode de minimisation suppose que les données de la vidéo du modèle soient structurées en un graphe ; par contre, cela n'est pas nécessairement demandé pour les données de la vidéo de la scène. Des informations structurelles sur les données de scène *peuvent* être intégrées facilement dans notre formulation, ce qui donne le problème classique d'appariement de graphes. Notre formulation est donc plus générale ; elle peut toutefois également traiter des problèmes d'appariement de graphes.

Selon l'hypothèse nr. 3, un appariement valide impliquera un appariement des frames du modèles aux frames de la scène. Pour cette raison nous reformulerons la fonction d'énergie (4.3) en divisant chaque variable x_i en deux variables z_i et $x_{i,l}$, qui seront intérprétées de manière suivante : la valeur de z_i est l'indice de la frame de la scène appariée avec la frame i du modèle. Le nombre de frames du modèle sera noté comme $\overline{M}$. Chaque frame i du modèle comprend un nombre $\overline{M}_i$

de variables $x_{i,1}, \ldots, x_{i,M_i}$, où la valeur $x_{i,l}$ correspond à l'indice du sommet auquel le sommet l de la frame i du modèle sera apparié dans la frame z_i de la scène. Ces indices seront numérotés en commençant par 1 dans *chaque frame de la scène*.

Afin de rendre la lecture plus facile, nous simplifierons la notation en représentant une hyper-arête (les indices de frame et les indices de sommet) par c et les variables correspondant comme z_c et x_c ; nous supprimerons également les poids λ_1 et λ_2 qui peuvent être absorbés dans les potentiels $U(.)$ et $D(.,.)$. Cela donne l'équation reformulée suivante :

$$E(z,x) = \sum_{(i,l)\in\overline{M}\times\overline{M}_i} U(z_i, x_{i,l}) + \sum_{c\in\mathcal{E}} D(z_c, x_c) \tag{4.16}$$

Ensuite nous introduisons une décomposition de l'ensemble des arêtes $\mathcal{E}$ en sous-ensembles disjoints $\mathcal{E}^i$, où $\mathcal{E}^i$ est l'ensemble d'arêtes c tel que c contient au moins un sommet avec une coordonnée temporelle égale à i et aucun sommet dans c a une coordonnée temporelle supérieure (donc, plus tard) à i :

$$\mathcal{E}^i = \Big\{ c \in \mathcal{E} : \max^{\langle t\rangle}(c) = i \Big\} \tag{4.17}$$

Ici $\max^{\langle t\rangle}(c)$ est la coordonnée temporelle maximale des sommets de l'arête c. Il est facile de voir que l'ensemble des $\mathcal{E}^i$ donne un partitionnement complet de $\mathcal{E}$, c.à.d. que $\mathcal{E} = \bigcup_i \mathcal{E}^i$. Nous pouvons maintenant échanger les sommes et le minima de notre problème selon ce partitionnement :

$$\begin{aligned}
&\min_{z,x} E(z,x) = \\
&\min_{z_1; x_{1,1},\ldots,x_{1,\overline{M}_1}} \quad \left[\sum_{l=1}^{\overline{M}_1} U(z_1, x_{1,l}) + \sum_{c\in\mathcal{E}^1} D(z_c, x_c) + \right. \\
&\min_{z_2; x_{2,1},\ldots,x_{2,\overline{M}_2}} \quad \left[\sum_{l=1}^{\overline{M}_2} U(z_2, x_{2,l}) + \sum_{c\in\mathcal{E}^2} D(z_c, x_c) + \right. \\
&\qquad\qquad\qquad\qquad \vdots \\
&\min_{z_{\overline{M}}; x_{\overline{M},1},\ldots,x_{\overline{M},\overline{M}_{\overline{M}}}} \quad \left[\sum_{l=1}^{\overline{M}_{\overline{M}}} U(z_{\overline{M}}, x_{\overline{M},l}) + \sum_{c\in\mathcal{E}^{\overline{M}}} D(z_c, x_c) \right] \ldots \left. \right]
\end{aligned} \tag{4.18}$$

Notre objectif est de dériver un algorithme récursif de minimisation en s'appuyant sur (4.18). Pour cela, nos introduisons le concept de la *portée* $\mathcal{R}^i$ de la frame i qui, intuitivement, est l'ensemble d'arêtes atteignant le passé de la frame i et qui viennent de i ou de son avenir ($> i$). Plus formellement,

$$\mathcal{R}^i = \Big\{ c \in \mathcal{E} : [\min^{\langle t\rangle}(c) < i] \wedge [\max^{\langle t\rangle}(c) \geq i] \Big\} \tag{4.19}$$

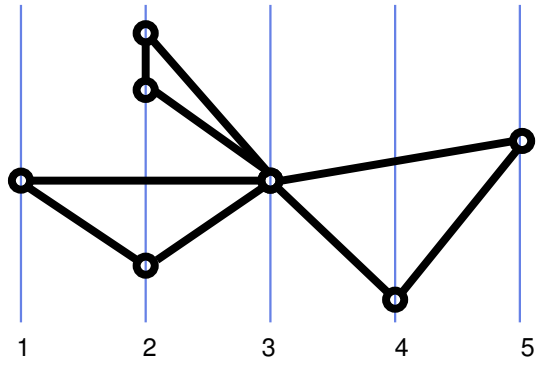
où $\min^{\langle t\rangle}(c)$ est définie de manière analogue à $\max^{\langle t\rangle}(c)$. Notons que $\mathcal{E}^i \subseteq \mathcal{R}^i$.

Nous introduisons également l'expression $\mathcal{X}^i$ pour l'ensemble de toutes les variables z_i et $x_{i,j}$ impliquées dans les arêtes de la portée $\mathcal{R}^i$:

$$\mathcal{X}^i = \big\{ z_j : \exists k : (j,k) \in c \wedge c \in \mathcal{R}^i \big\} \cup \big\{ x_{j,k} : (j,k) \in c \wedge c \in \mathcal{R}^i \big\} \tag{4.20}$$

Enfin, les variables $\mathcal{R}^i$ restreints aux variables des frames *avant* la frame i seront dénotés comme $\mathcal{X}^{i-}$:

$$\mathcal{X}^{i-} = \{ z_j, x_{j,k} \in \mathcal{X}^i : j < i \} \tag{4.21}$$



$$
\begin{aligned}
\alpha_5(3,4) &= \min_5[\ D(3,4,5)] \\
\alpha_4(3) &= \min_4[\ \alpha_5(3,4)] \\
\alpha_3(1,2) &= \min_3[\ D(1,2,3)+ \\
&\qquad\qquad D(2,2,3)+\alpha_4(3)] \\
\alpha_2(1) &= \min_2[\ \alpha_3(1,2)]
\end{aligned}
$$

Figure 4.16 – (a) un exemple de hyper-graphe — les barres verticales sont les frames de la vidéo ; (b) la décomposition de la fonction d'énergie dans une notation très simplifiée : les arguments et variables indiquent les numéros de frames ; seulement les termes ternaires ont été inclus.

Le schéma de calcul récursif minimisant (4.18) peut maintenant être dérivé en définissant une variable récursive α_i contenant la minimisation des variables d'une frame donnée étant données les valeurs optimales pour les variables de sa portée :

$$\alpha_i(\mathcal{X}^{i^-}) = \min_{z_i; x_{i,1},\ldots,x_{i,\overline{M}_i}} \left[\sum_{l=1}^{\overline{M}_i} U(z_i, x_{i,l}) + \sum_{c \in \mathcal{E}^i} D(z_c, x_c) + \alpha_{i+1}(\mathcal{X}^{(i+1)^-}) \right] \tag{4.22}$$

Cette récursion nécessite que la relation suivante soit satisfaite :

$$\mathcal{X}^{(i+1)^-} \subseteq (\mathcal{X}^{i-} \cup z_i; x_{i,1}, \ldots, x_{i,\overline{M}_i}). \tag{4.23}$$

Preuve — Les membres gauche et droit de l'équation (4.23) impliquent des arêtes des deux ensembles R^{i+1} et R^i. De l'équation (4.19) nous pouvons déduire que les seuls triangles faisant partie de R^{i+1} et non pas de R^i sont les triangles dont la coordonnée temporelle minimale est i. Il se pose la question s'il existe des variables associées à ces triangles et qui font partie de $\mathcal{X}^{(i+1)^-}$ mais pas de $(\mathcal{X}^{i-} \cup \{z_i; x_{i,1}, \ldots, x_{i,\overline{M}_i}\})$. D'après l'équation (4.21), l'ensemble $\mathcal{X}^{(i+1)^-}$ ne contient aucune variable associée à une frame supérieur à i, cela concerne donc uniquement les variables de la frame i même, qui ont été inclues dans le membre droit de l'équation (4.23) de manière explicite par l'opérateur d'union. □

Le calcul de la récursion est initié pour la dernière frame $i = \overline{M}$. Ensuite l'algorithme procède de manière itérative en calculant α_i à partir de α_{i+1}. A chaque itération, un minimum est calculé pour toutes les variables de la frame i pour toutes les valeurs possibles des variables de $\mathcal{X}^{i^-}$. La complexité de calcul dépend donc du nombre de variables dans la portée $\mathcal{R}^i$ et des tailles des domaines de ces variables :

$$O\left(\max_i \left[\prod_{v \in \mathcal{V}^i} |\text{domain}(v)| \right] \right) \approx O\left(\max_i \left[\overline{S}^{|\mathcal{X}_z^i|} \langle\langle s \rangle\rangle^{|\mathcal{X}_x^i|} | \right] \right) \tag{4.24}$$

où $\overline{S}$ est le nombre de frames de la scène, $\langle\langle s \rangle\rangle$ est le nombre moyen de sommets par frame dans le modèle et $|\mathcal{X}_z^i|$ est le nombre de variables de l'ensemble z dans $\mathcal{X}^i$. La complexité est donc très inférieure à la complexité de l'approche directe, donnée par $O(S^M M|\mathcal{E}|)$. Notons que S est

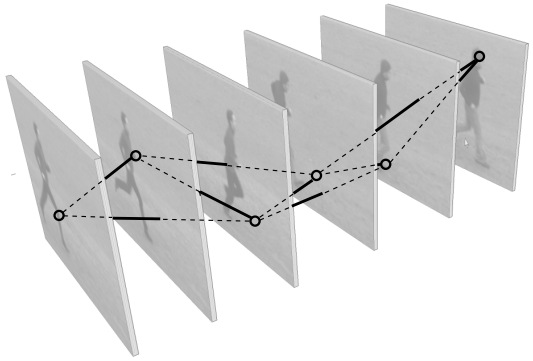


Figure 4.17 – Une structure graphique pour le modèle, construite pour une faible complexité de calcul : une chaîne de deuxième ordre. Aucune structure est imposée sur les points de la scène.

le nombre total de sommets dans la scène et M est le nombre total de sommets du modèle ; surtout, $S \gg \overline{S}$ et $S \gg \langle\langle s \rangle\rangle$. Aussi, $|\mathcal{X}_z^i|$ et $|\mathcal{X}_x^i|$ sont bornés et faible si le graphe est contruit à partir d'informations de proximité. Par contre, en pratique la complexité reste assez élevée. Dans la sous-section suivante nous introduirons une structure graphique spécifique pour rendre l'algorithme encore plus efficace.

4.6.4 Un modèle de deuxième ordre

La plupart des formulations du problème d'appariement de graphes ou d'appariement à partir d'un graphe sont NP-complets. Les méthodes classiques résolvent ce problème en calculant une solution approchée. Ici, nous préconisons une autre idée, à savoir approcher le problème ou lieu de l'algorithme d'inférence. Dans ce cas, nous proposons d'approximer la structure graphique et de résoudre le nouveau problème de manière exacte. Cette stratégie est particulièrement attrayante dans le cas des problèmes d'appariement où la structure du graphe est moins liée à la description de l'objet, mais plutôt aux contraintes du processus d'appariement. Nous rappelons que la structure graphique est obtenue à partir des informations d'adjacence ou de proximité. Elle est donc déduite des attributs des sommets (les positions), la changer ne nuira pas sensiblement à la description de l'objet spatio-temporel. Une philosophie similaire a été mis en avant par [TCSB06] dans le contexte de la reconnaissance d'objets, où l'objet spatial a été structuré en un k-arbre, rendant peu complexe l'algorithme *junction tree* utilisé pour la minimisation de la fonction d'énergie.

Nous proposons de structurer les points du modèle comme suit :

— Un seul point unique est gardé pour chaque frame du modèle en choisissant le point le plus saillant, c.à.d. le point choisi avec une confiance maximale par le détecteur de points d'intérêt spatio-temporels. Aucune restriction est placée sur la scène. En particulier, chaque frame de la scène peut contenir autant de points que souhaité.

— Chaque point i du modèle est connecté à ses prédécesseurs immédiats $i{-}1$ et $i{-}2$ et à ses successeurs immédiats $i{+}1$ et $i{+}2$.

Cela donne un graphe planaire avec une structure triangulaire comme illustré dans la figure 4.17. La notation de l'énergie peut être simplifiée pour tenir compte de cette structure. La division des variables en paires (z_i, x_i), introduite dans la section 4.6.3, n'est plus nécessaire. Le système de

	B	HC	HW	J	R	W
B	**100**	0	0	0	0	0
HC	0	**100**	0	0	0	0
HW	3	26	**71**	0	0	0
J	0	0	0	**69**	31	0
R	0	0	0	25	**75**	0
W	0	0	0	3	3	**94**

(a)

	B	HC	HW	J	R	W
B	**100**	0	0	0	0	0
H	3	**97**	0	0	0	0
H	6	15	**79**	0	0	0
J	0	0	0	**72**	28	0
R	0	0	0	8	**89**	3
W	0	0	0	6	0	**100**

(b)

Table 4.5 – La matrice de confusion avec (a) et sans (b) sélection de modèles. Les taux de reconnaissance respectifs : 84.8%, 89.3%. (B : Box, HC : Handclap, HW : Handwave, J : Jog, R : Run, W : Walk).

voisinage peut être décrit de manière très simple en se basant sur les indices des variables x_i :

$$E(x) = \sum_{i=1}^{M} U(x_i) + \sum_{i=3}^{M} D(x_i, x_{i-1}, x_{i-2}) \tag{4.25}$$

La portée de cette structure est constante, elle consiste de deux arêtes par frame i : $\mathcal{R}^i = \{(x_{i-2}, x_{i-1}, x_i), (x_{i-1}, x_i, x_{i+1})\}$; L'ensemble des variables de la portée est également constant : $\mathcal{X}^i = \{x_{i-2}, x_{i-1}, x_i, , x_{i+1}\}$. La récursion peut être donnée par l'équation suivante :

$$\alpha_i(x_{i-1}, x_{i-2}) = \min_{x_i} \Big[U(x_i) + D(x_{i-2}, x_{i-1}, x_i) + \alpha_{i+1}(x_i, x_{i-1}) \Big] \tag{4.26}$$

L'algorithme sous la forme donnée ci-dessus à une complexité de calcul de $O(M{\cdot}S^3)$: un treillis est calculé à l'aide d'une matrice de taille $M \times S \times S$, où le calcul de la valeur de chaque cellule demande une itération sur S valeurs. En profitant des différentes hypothèses introduites dans la section 4.6.1, la complexité peut encore être diminuée :

Ad) Hypothesis 1 — en tenant compte des contraintes de causalité, un grand nombre de combinaisons peuvent être supprimées du treillis. Pour une valeur donnée de x_i, les valeurs des prédécesseurs x_{i-1} et x_{i-2} doivent être *avant* x_i, c.à.d. inférieures.

Ad) Hypothesis 2 — de manière similaire, nous imposons une restriction importante aux valeurs de x_{i-1} et x_{i-2} : elles sont supposées être proches, c.à.d. la distance doit être inférieure à T^t.

Cela diminuera la complexité à $O(M{\cdot}S{\cdot}T^{t^2})$, où T^t est une petite constante (autour de 15) ; la complexité est donc linéaire en fonction du nombre de frames de la scène : $O(M{\cdot}S)$.

4.6.5 Expériences

Nous avons testé la méthode proposée sur la base de vidéos KTH [SLC04], une base publique et largement utilisée. Elle comprend 25 personnes effectuant 6 actions (*walking*, *jogging*, *running*, *handwaving*, *handclapping* et *boxing*) enregistrées dans quatre différentes scénarios, y compris des scènes à l'intérieur, à l'extérieur et des points de vue de caméra différents.

Les classes des vidéos de la base de test sont reconnues avec un classifier « plus proche voisin » (NN) avec comme distance l'énergie de l'appariement. Un dictionnaire optimal et équilibré a été

Method	B	HC	HW	J	R	W	Tot.
Laptev *et al.* [LMSR08]	97	95	91	89	80	99	91.8
Schuldt *et al.* [SLC04]	98	60	74	60	55	84	71.8
Li *et al.* [LAM+11]	97	94	86	**100**	83	97	**92.8**
Niebles *et al.* [NCFF10]	99	**97**	**100**	78	80	94	91.3
Notre méthode	**100**	**97**	79	72	**88**	**100**	89.3

Table 4.6 – Comparaison avec des méthodes existantes utilisant le même protocole et la même base de vidéos (KTH). (B : Box, HC : Handclap, HW : Handwave, J : Jog, R : Run, W : Walk).

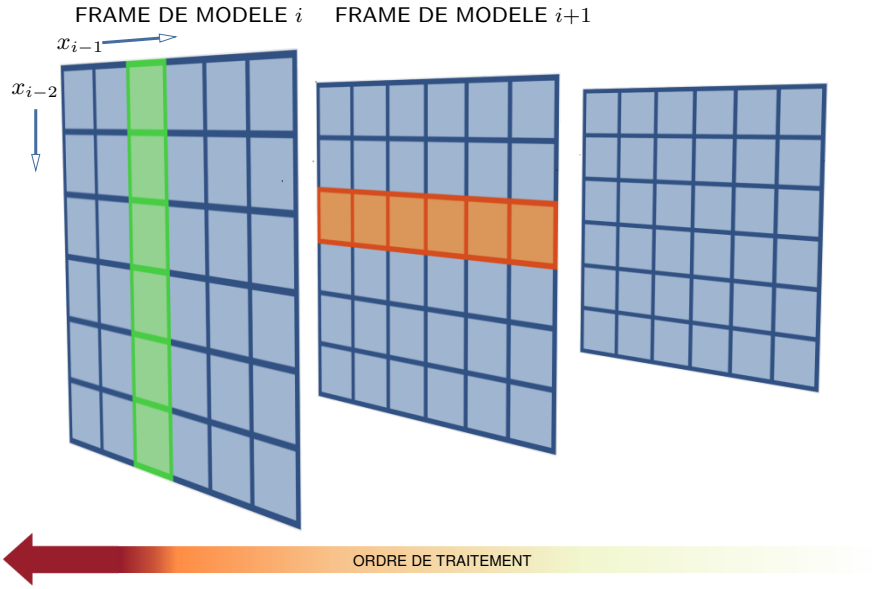


Figure 4.18 – La distribution du traitement sur les unités d'un GPU permet un calcul bien plus rapide que le temps réel.

obtenu avec *Sequential Floating Backward Search* (SFBS), une procédure qui supprime des modèles non pertinents de l'ensemble d'apprentissage [PFNK94].

Dans la figure 4.19 quelques exemples d'appariement sont donnés : les premiers deux cas sont des exemples pour un appariement correct ; les 4é cas est un exemple pour un appariement incorrect. La table 4.6 montre que la performance de notre méthode se compare avec les méthodes de l'état de l'art, tout en étant beaucoup plus que compétitive en terme de temps de calcul — nous procédons 5 fois plus vite que le temps réel. Nous voudrons souligner que des nombreux résultats ont été publiés sur la base de données KTH. Par contre, les protocoles ne sont pas comparables pour la plupart d'entre eux — voir l'excellente comparaison des protocoles dans [GCHA10]. Nous avons choisi de montrer les résultats obtenus avec le même protocole (le protocole d'origine propose avec la base de vidéos dans [SLC04]) dans la table 4.6.

La méthode a été implémentée en une version parallèle avec un traitement par GPU (carte graphique) par Eric Lombardi du LIRIS. La figure 4.18 illustre comment le traitement est distribué sur les unités de calcul de la GPU. La recursion de l'équation (4.26) traite un trellis 3D qui peut être

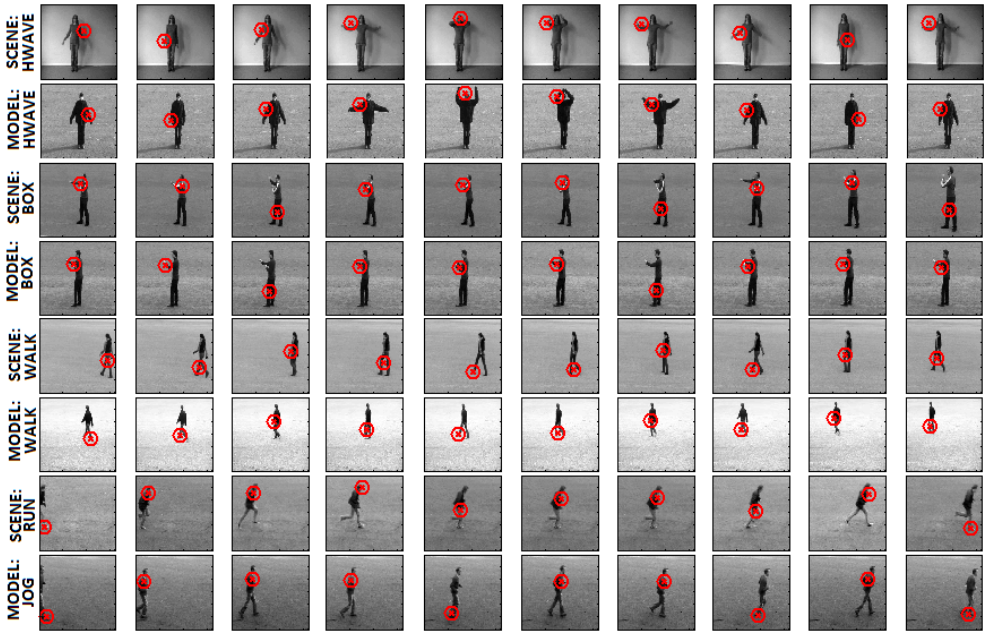


Figure 4.19 – Résultats de classification pour la base KTH.

représenté comme une séquence de tableaux 2D, chacun correspondant à une frame du modèle. Les tableaux 2D sont traités de manière séquentielle, et les cellules d'un tableau sont calculées en parallèle. Un group de travail de la carte traite une colonne du tableaux, montrée en vert dans la figure 4.18. Le calcul de ce groupe dépend d'une ligne de la frame traitée précédemment, montrée en orange dans la figure 4.18. La mémoire locale d'un groupe contient donc une ligne et une colonne de la structure globale.

L'implémentation sur carte graphique permet un traitement bien plus rapide que le temps réel. En découpant chaque vidéo en blocs de 60 frames, l'appariement exact et optimal entre une vidéo de scène et les 46 graphes de modèle choisis par la méthode SFBS demande 8.28ms par frame, donc 0.2 secondes de calcul par seconde de vidéo.

4.7 Conclusion

Dans ce chapitre nous avons présenté nos travaux sur la modélisation de formes visuelles complexes par modèles structurés et semi-structurés. Nos contributions principales traitent de l'appariement de graphes et par graphes.

Nous avons montré que — lorsque les données sont plongées dans l'espace-temps — la solution exacte du problème d'appariement à l'aide d'un hyper-graphe peut être calculée en une complexité de calcul exponentielle sur un petit nombre, qui est borné lorsque l'hyper-graphe a été structuré avec des informations de proximité. Dans ce cas la complexité est donc polynomiale. Nous avons présenté une structure de graphe spécifique permettant de calculer l'appariement exact avec une complexité très faible, linéaire dans le nombre des sommets du modèle et dans le nombre de sommets de la scène.

Nous sommes en train de continuer nos recherches dans cette thématique, et de nombreuses

perspectives s'ouvrent :

Méthodes exactes nous sommes en train de travailler sur un modèle similaire au modèle de type 2^e ordre introduit dans la sous-section 4.6.4. Le modèle permettra un nombre arbitraire de sommets par frame, tout en garantissant une décomposition simple de la fonction d'énergie, permettant la conception d'algorithmes efficaces.

Méthodes spectrales Des travaux actuellement en cours traitent l'intégration des propriétés spécifiques de l'espace-temps (voir la sous-section 4.6.1) dans les méthodes spectrales classiques, permettant de les rendre plus performantes.

Segmentation par la profondeur Grâce aux capteurs tels que MS Kinect, les images et vidéos en profondeur (dite de 3D) sont maintenant disponibles pour certaines situations (intérieur, pièces pas trop grandes etc.). Nous prévoyons de travailler sur des algorithmes de segmentation de scènes et sur les présentations par graphes de scènes et de leurs acteurs.

Modèles génératifs Dans le cadre d'une thèse débutant en octobre 2012 et co-encadrée par Gérard Bailly, DR au Gipsalab, Grenoble, nous travaillerons sur l'apprentissage de modèles d'interactions entre hommes et robots. Il s'agit de modèles génératifs permettant de reproduire les actions appris par le système.

Nous renvoyons le lecteur au chapitre 7 pour une description plus générale des perspectives de nos travaux.

Chapitre 5

Segmentation et restauration d'images et de vidéos

Dans ce chapitre nous discuterons de la segmentation d'images, un problème que les communautés de traitement d'images et de vision par ordinateur traitent depuis plusieurs décennies. Or, même après une grande quantité de travaux effectués, de nombreux défis restent encore ouverts. Les activités autour de cette thématique sont attestées par les importantes participations aux compétitions scientifiques, comme par exemple la série DIBCO 2009, HDIBCO 2010 et DIBCO 2011 — *"(Handwritten) Document Image Binarization Competition"*[1]. D'une part, cela s'explique par les projets de numérisation de livres de grande envergure qui sont actuellement effectués par plusieurs organisations, comme, entre autres, Google. Cette numérisation est accompagnée d'une reconnaissance automatique nécessitant une binarisation préalable[2]. D'autre part, le patrimoine mondial contient également un grand nombre de documents très anciens, qui ne peuvent être reconnus facilement de manière automatique. Par contre, à cause de leur souvent piètre état, une restauration numérique s'avère nécessiare, un traitement qui est souvent basé sur une phase de segmentation afin de pouvoir identifier des zones à éliminer : le verso apparaissant sur le recto, des tâches, des trous etc.

5.1 Contexte, projets et collaborations

La grande majorité des travaux décrits dans ce chapitre sont des travaux personnels. Les premières recherches ont débuté assez tôt après notre intégration dans le groupe de l'analyse de documents de l'équipe « Imagine » du LIRIS après notre mutation à Lyon en 2005. Les méthodes traditionnellement employées dans le groupe étant basées sur le clustering [LLE04, DLE06b, DLE06a] et sur les EDP [DLE07, DL11], nous étions (et nous sommes toujours) convaincus de l'efficacité d'une modélisation Markovienne dans ce contexte. Les travaux sur le modèle de Markov à double couche ont été soumis (dans une première version) en 2006 [Wol06]. Suite à un problème d'organisation

1. Nous avons d'ailleurs participé à la compétition DIBCO 2009 avec un résultat assez intéressant : notre algorithme a obtenu la 5^e place sur 43 participants. L'algorithme de binarisation, publié dans [WJC02], ne sera pas présenté dans ce mémoire.

2. A long terme, la recherche en reconnaissance automatique de caractères se tournera sans doute vers le traitement direct d'images en niveaux de gris ou en couleur. Par contre, pour le moment, la plupart des moteurs de reconnaissance passent par une binarisation de l'image.

interne du journal choisi [3], nous avons publié l'article dans un journal différent avec un retard assez conséquent de plusieurs années [Wol10].

Les travaux sur le modèle hiérarchique « cube de Markov » ont été inspirés par les travaux sur l'arbre quaternaire probabiliste [LPH00] de l'équipe MIV du laboratoire LSIIT à Strasbourg, notre laboratoire d'accueil en 2004-2005. Fasciné par l'élégance des algorithmes d'inférence et d'apprentissage de ce modèle, nous nous sommes attaqué à résoudre ses faiblesses, à savoir sa non stationnarité et les artefacts de segmentation qui en découlent. Dans le cadre de ces travaux nous avons collaboré avec Gérald Gavin du laboratoire ERIC [4] sur l'interprétation des probabilités conditionnelles et sur la triangulation du graphe de dépendances du modèle.

Les travaux de ce chapitre ont été effectués sans le contexte d'un projet de recherche particulier, c.à.d. sans aucun financement particulier. La seule exception sont les recherches sur la soustraction de fond dans la vidéos décrits dans la section 5.8. En collaboration avec Jean-Michel Jolion du LIRIS, ces travaux ont été poursuivis dans le cadre du projet ANR Canada sur la détection d'évènement anormaux — voir le chapitre 4 pour plus d'informations sur ce projet.

Nous terminerons ce chapitre avec une brève description de nos travaux sur la segmentation d'une image du corps humain en parties. Destinés à l'estimation de la pose, ces travaux ont été effectué dans le cadre de la thèse de Mingyuan Jiu co-encadré par Atilla Baskurt.

5.2 Modèles graphiques et segmentation d'images

Ici, nous préconisons l'usage de modèles graphiques pour les applications mentionnées ci-dessus, qui se proposent comme un choix naturel. En effet, la segmentation d'une image correspond à une classification des pixels d'une image en plusieurs classes, avec comme objectif le regroupement des pixels de même classe en régions. Deux types d'informations sont souvent traités :

- la couleur ou le niveaux de gris local, par exemple au niveau d'un pixel. Cette information peut donner une indication directe de l'appartenance d'un pixel en question à une classe : texte, fond, tâche, texte du verso etc. ;
- les résultats de classification de pixels voisins, qui sont inconnus. Les configurations locales des résultats et leur géométrie peuvent être prises en compte dans la segmentation, aboutissant à des segmentations favorisées par une connaissance *a priori* sur le contenu. Cela va créer des dépendances circulaires qui sont naturellement faciles à modéliser par des modèles graphiques probabilistes.

Une fois le framework des modèles graphiques probabilistes choisi, les informations colorimétriques peuvent être modélisées par les termes d'attaches aux données, donc les termes associés aux cliques de taille 1. Les informations géométriques sont logiquement modélisées par des termes de régularisation, donc des termes associés aux cliques de taille supérieure. Il reste à choisir le type de modèle, génératif ou discriminatif, et la structure du graphe.

Dans ce chapitre, nous proposerons plusieurs modèles et algorithmes destinés à plusieurs applications. Le point commun est le problème sous-jacent : il s'agit d'un problème de segmentation d'images de documents. Dans tous les cas, l'image est supposée contenir du texte, donc un ensemble de caractères. Or, le contexte et les objectifs peuvent s'avérer légèrement différents selon

3. Notre soumission a été oubliée dans le système électronique et nous avons en conséquence retiré l'article pour le soumettre à un autre journal.

4. Au moment de la collaboration, Gérald Gavin était membre de l'équipe MA2D du laboratoire LIRIS.

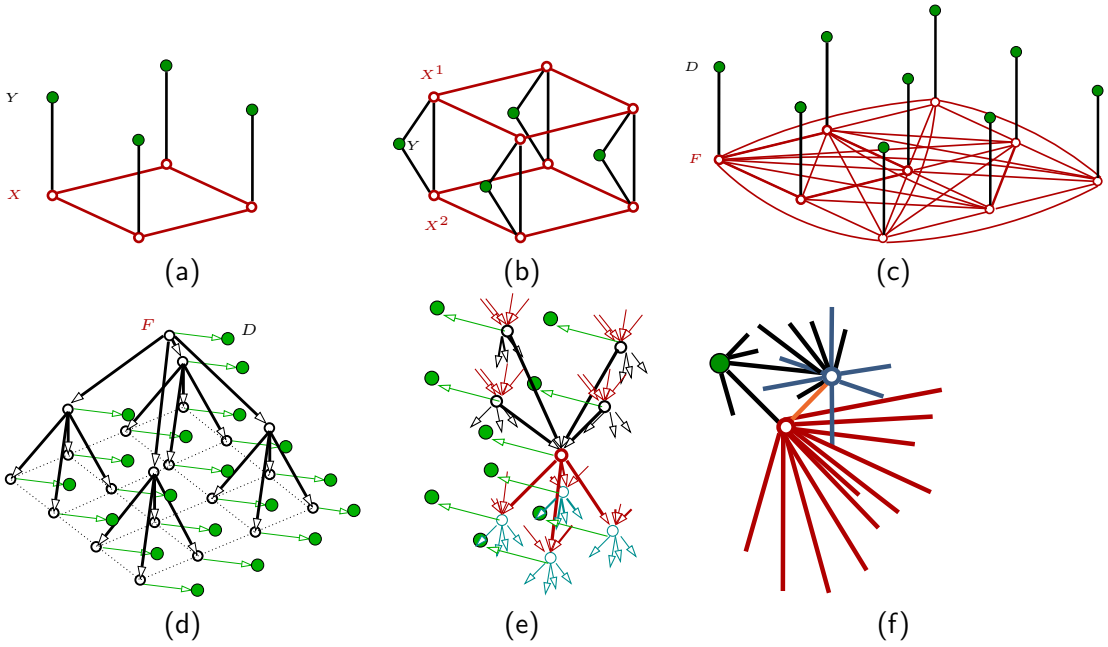


Figure 5.1 – Quelques modèles graphiques adaptés à la ségmentation d'images ou de vidéos : (a) un MRF classique ; (b) un MRF à double couche pour la restauration recto/verso d'un document [Wol10] (voir la section 5.3) ; (c) un MRF à cliques larges pour la restauration de caractères [WD02] ; (d) un arbre de Markov classique [LPH00] ; (e) un extrait d'un cube de Markov hiérarchique pour la restauration de documents [WG10] (voir la section 5.4) ; (f) un MRF à plusieurs types de variables cachées pour la segmentation fond-forme de vidéos conjointement avec l'estimation de mouvement [WJ10] (voir la section 5.8).

le contexte :

- la binarisation d'une image de document numérisé à une résolution importante. Il s'agit donc d'une segmentation en deux classes ;
- la restauration numérique d'un document ayant subis une dégradation de type mélange de verso avec le recto. Il s'agit de supprimer le contenu du verso paraissant sur le texte du recto ;
- la binarisation d'images acquises en faible résolution, nécessitant une restauration numérique de caractères afin de pouvoir les reconnaitre.

Nous plaidons pour l'adaptation de la structure du modèle graphique au problème en question, ç.à.d. aux conditions d'acquisition, à l'application spécifique, aux dégradations, aux connaissances *a priori* disponibles etc. Nous montrerons qu'il est préférable de choisir la structure du modèle en fonction de critères très simples, comme par exemple la résolution de numérisation ; et que cela peut renforcer considérablement les résultats de classification et, *a fortiori*, de reconnaissance automatique de caractères.

Les choix de structures de graphe classiques employées dans ce contexte ont été guidés par les objectifs généraux du problème de segmentation, à savoir le partitionnement d'une image en régions homogènes et sans chevauchement. Les réseaux Bayesiens et les MRF ont très fréquemment été utilisés pour inclure les dépendances spatiales entre les pixels dans le processus de classification, souvent formulé par un problème d'estimation Bayésienne. Dans leur celebre papier [GG84],

Geman et Geman introduisent une technique d'estimation a posteriori (MAP) pour les MRF. Cette structure, dorénavant classique, comprend un champ caché et un champ observé. La structure très simple est illustrée dans la figure 5.1a.

Une alternative aux MRF non causaux à deux dimensions sont les chaînes de Markov cachées (HMM), pourtant définies sur une seule dimension. Leur application aux images est possible grâce au parcours de Hilbert-Peano, un parcours fractal des pixels d'une image [AHL65, PT00, FDP+03, SC06] — voir la figure 3.7 dans le chapitre 3. La structure causale et l'absence de cycle dans le graphe de dépendances permet d'obtenir des algorithmes d'inférence très peu complexes pour des fonctionnelles d'énergie arbitraires, notamment non sous-modulaires. L'avantage est une prise en compte approximative des relations de voisinage : l'ensemble des voisins de chaque site est restreint à deux. Des techniques hybrides HMM/MRF ont été présentées [FDP+03]. Les modèles à base de chaînes de Markov ont été étendus aux réseaux Bayesiens avec une structure approchant une grille 2D [KA94, LP92].

L'objectif des modèles hiérarchiques est d'introduire une composante dépendant de l'échelle, ce qui permet à l'algorithme de mieux s'adapter aux caractéristiques de l'image. L'ensemble des sommets du graphe est partitionné en différentes échelles : les niveaux de l'échelle inférieure correspondent aux versions plus fines de l'image et les niveaux plus élevés correspondent à des versions plus grossières de l'image. Au niveau bas, les interactions entre les pixels sont gérées, alors que des échelles supérieures traitent les interactions entre les groupes de pixels, ç.à.d. entre les régions. Plusieurs types de modèles hiérarchiques ont été introduits : des piles de modèles plats de type MFR [Bel94], des structures pyramidales [KBZ96] et la multi-grille causale en échelle [MCPB00].

Bouman et Shapiro ont été parmi les premiers à proposer un réseau Bayesien hiérarchique pour la segmentation d'images [BS94], une méthode plus tard raffinée par Laferte et al. [LPH00]. Un arbre quaternaire lie chaque pixel de l'image à une racine commune en passant par des sommets intermédiaires correspondant à des régions de plus en plus grandes — un exemple est montrée dans la figure 5.1d. L'avantage principale du modèle est sa causalité et l'absence de cycles dans le graphe, ce qui rend l'inférence très efficace. L'inconvénient principal est la non stationnarité induite dans le champ aléatoire marginal correspondant aux feuilles de l'arbre, donc les pixels de l'image. Cela peut produire des artefacts ressemblant à des blocs dans l'image segmentée. Le modèle a également été utilisé pour la segmentation multi-d'images spectrales [PCR+04].

Dans le même travail un deuxième modèle est proposé, où chaque sommet a trois parents. A première vue, la structure du graphe de dépendances est un peu similaire à la solution que nous décrirons dans la section 5.4 (qui dispose de quatre parents pour chaque site). Cependant, le modèle proposé par Bouman est un modèle pyramidal où le nombre de sommets diminue à chaque niveau, alors que dans notre modèle le nombre de sommets est le même pour tous les niveaux. Par ailleurs, pour rendre l'inférence dans le modèle facile, l'algorithme d'inférence de Bouman et al. change la structure du graphe à chaque itération, ç.à.d. pour chaque niveau, alors que dans notre méthode l'ensemble du graphe conserve sa connectivité complète.

Une segmentation ou une restauration performante d'images, et plus particulièrement de documents, nécessite la modélisation la plus exacte possible du processus de dégradation, ainsi que du contenu des documents, si cette information est disponible. Afin de pouvoir facilement intégrer toutes les connaissances applicatives, nous avons choisi la famille de modèles génératifs. Dans ce cadre Bayesien, les informations disponibles peuvent être intégrées dans le processus de restauration en tant qu'informations *a priori* ou dans le modèle d'observation. Nous avons amélioré l'état de l'art dans ce domaine en proposant plusieurs contributions :

— pour le cas spécifique de la séparation recto/verso, nous avons proposé un modèle probabiliste de type MRF à double couche gérant les interactions entre les deux cotés cachés d'un document et de la seule face visible [Wol10, Wol08] — voir la figure 5.1b. Le modèle sera détaillé dans la section 5.3.
— pour le cas général, et plus particulièrement pour des images de documents numérisés en grande résolution, un nouveau modèle hiérarchique de type réseau Bayesien a été proposé capable de gérer les interactions spatiales sur différents niveaux de manières différentes [WG10] — voir la figure 5.1d. Le modèle sera détaillé dans la section 5.4.
— pour les deux modèles, des algorithmes performants d'optimisation discrète à base de *graph cuts* ont été proposés afin d'obtenir une excellente solution proche de l'optimum global [Wol10, WG10]. Plus de détails seront données dans la section 5.5.
— pour la binarisation d'images acquises en faible résolution, un modèle de type MRF à larges cliques a été proposé [WD02] — voir la figure 5.1c. Nous mentionnons ces travaux moins récents ici, ils ne seront pas décrits dans ce mémoire.

La figure 5.1 montre les différents graphes de dépendances des modèles développés dans ce chapitre, ensemble avec quelques graphes de dépendances classiques. Ces modèles, ainsi que les algorithmes associés, seront plus détaillés dans les sections suivantes : les sections 5.3 et 5.4 introduiront les modèles développés. La section 5.5 donnera des détails sur la minimisation des différentes énergies associées aux modèles. La section 5.6 abordera l'estimation des paramètres de ces modèles. Et la section 5.7 présentera quelques expériences et les résultats.

La section 5.8 traite un problème très similaire d'un point de vue scientifique, issue d'une application différente. Il s'agit de la séparation fond/forme dans les flux vidéos, qui relève de la détection des objets en mouvement. Nous verrons que ce problème peut se formaliser de manière très similaire en un problème de binarisation d'images impliquant des termes d'attache aux données et des termes de régularisation. Nous montrerons, que l'ajout de termes issus d'une estimation de mouvement améliore les résultats.

La dernière section, section 5.9, présentera nos travaux sur la segmentation d'images sans régularisation combinatoire. L'algorithme présenté recourt uniquement à un classifieur pour prendre les décisions de manière indépendante pour chaque pixel.

5.3 Séparation recto/verso d'images de document

Notre premier modèle traite le problème de la suppression de la face verso d'un document visible dans une image après la numérisation de la face recto. Nous supposons que la numérisation de la face verso n'est *pas disponible* (séparation aveugle). Dans ce cas, la tâche peut se résumer à un problème de segmentation : classification de chaque pixel en tant que *recto, verso, fond*, et éventuellement *recto-et-verso* (en même temps). Cela rend immédiatement disponible la vaste collection de techniques de segmentation largement connues. Cependant, les images de documents sont un type particulier d'images avec leurs propres propriétés et de leurs propres problèmes.

Plusieurs techniques ont été proposées dans la littérature pour ce type de problèmes. Il peut se formuler de manière évidente comme un problème de séparation aveugle de sources, semblables aux problèmes de type *cocktail party* traités avec succès par la communauté de traitement du signal. Une technique largement utilisée, l'analyse en composantes indépendantes (ICA), a été appliquée aux documents, principalement par Tonazzini et al. [TB04]. Toutefois, l'ICA repose sur un modèle

linéaire qui n'est pas justifié dans le cas des documents. Au contraire, selon nos expériences, l'encre du recto a plutôt tendance à cacher l'encre du verso (hypothèse d'opacité), ce qui est impossible à modéliser par une interaction linéaire. Les mêmes auteurs ont également introduit une technique non-aveugle applicable aux images en niveaux de gris [TSB07], les différentes composantes correspondant aux pages recto et verso. Dans [TBS06] les mêmes auteurs introduisent un modèle de MRF à double couche, similaire à notre proposition, combiné avec un terme de vraisemblance composé d'un modèle de mélange linéaire. Cependant, alors que notre modèle graphique est directement utilisé pour la classification, le MRF dans [TBS06] sert à guider un algorithme de type *expectation maximization* (EM) pour estimer l'inverse de la matrice de mélange. Comme avec les autres algorithmes basés sur le mélange, la plus grande faiblesse est la linéarité du modèle. Dans [TG05], le modèle est étendu aux mélanges convolutifs.

D'autres méthodes dans l'état de l'art se servent d'une estimation de l'orientation locale pour la séparation, soit par approches fréquentielles [WXTL03, NS02] soit par EDP [DLE07, DL11]. Des approches non-aveugles nécessitant l'alignement des pages ont été présentées [Sha01, DP01, TCS02], ainsi que des méthodes de seuillage sophistiquées et guidées par modèle [Don00]. La régularisation par approche variationnelle est possible — voir aussi nos remarques dans la section 6.2.2 du chapitre 6 sur les modèles géométriques.

La régularisation par champ de Markov (MRF) a déjà été appliquée à ce problème [TVB03, DM05, WD02]. Elle permet de créer un modèle statistique exploitant des connaissances sur le processus de dégradation ainsi que sur le contenu de l'image (la connaissance *a priori*). Toutefois, les méthodes précédentes traitent la séparation recto/verso de la même manière que la segmentation classique d'images. Dans ce travail, nous montrerons que les performances peuvent être améliorées de façon significative si les le problème est spécifiquement modélisé comme un problème de séparation de deux faces. Nous formulons notre méthode dans un cadre Bayésien impliquant deux informations :

— la connaissance *a priori* sur le document segmenté est modélisée par un MRF. Dans notre cas, il s'agit de deux champs, un pour chaque côté du document.
— le modèle d'observation modélise les connaissances sur le processus de dégradation des documents.

Les approches de segmentation par MRF sont souvent motivées par des hypothèses d'homogénéité, ç.à.d. des zones homogènes sont considérées comme étant plus probables que des changements fréquents d'étiquettes entre pixels voisins. Ce n'est pas justifié à tous les endroits de l'image observée lorsque celle-ci est le résultat d'une superposition de deux ou plusieurs "sources". Dans ce cas, les sources étant indépendantes l'une de l'autre, la connaissance *a priori* peut être disponible pour chaque source de l'image, mais pas pour le mélange.

Pour cette raison nous proposons un modèle *a priori* ayant deux champs d'étiquettes : un pour le côté recto (X^1) et un pour le coté verso (X^2), résultant en un modèle avec deux variables cachées pour chaque pixel (*recto* et *verso*) et un espace de configurations de deux valeurs pour chaque variable cachée (*texte* et *fond*). Les avantages de cette formulation sont les suivants :

— la connaissance *a priori* est appliquée à la partie du modèle pour laquelle elle a été obtenu, ç.à.d. pour les sources du mélange. Les hautes fréquences introduites par le mélange ne sont pas lissées par la régularisation markovienne ;
— une estimation des pixels verso recouverts par des pixels recto de type "texte" est possible, uniquement grâce à la modélisation par deux champs d'étiquettes séparés. Cette estimation n'est pas seulement souhaitable dans le cas où une estimation de la page verso est demandée

par l'application. Plus encore, une estimation correcte des pixels verso, par l'intermédiaire des interactions spatiales encodées par les deux champs d'étiquettes, aide à estimer correctement les pixels verso non couverts par un pixel recto de type "texte". Cela améliore les performances de l'algorithme en améliorant l'estimation de la face recto.

Nous tenons à préciser que le même résultat pourrait être atteint avec un champ d'étiquettes cachées unique et en adaptant le modèle tel que sa régularisation gère les interactions entre étiquettes de manière différente. Cependant, cela demanderait des fonctionnelles d'énergie plutôt compliquées équivalent à des interactions assez simples pour le cas d'une modélisation séparée. Par ailleurs, la formulation de l'algorithme d'inférence (la minimisation de l'énergie associée) aurait été bien plus complexe.

Traditionnellement les termes d'attache aux données et les termes de régularisation sont formulés de manière séparée. Étant donnée la nature de notre problème, nous préférons interpréter l'ensemble complet des variables cachées et observées comme un seul MRF. Dans la suite nous considérons donc un graphe $\mathcal{G} = \{V, E\}$ avec un ensemble de sommets V et un ensemble d'arêtes E, où V est partitionné en trois sous ensembles distincts : les deux champs cachés X^1 et X^2 et un champ observé Y correspondant à l'image d'entrée. Les trois champs sont indexés par les mêmes indices correspondant aux pixels de l'image, ç.à.d. X_i^1, X_i^2 et Y_i désignent, respectivement, l'étiquette cachée recto, l'étiquette cachée verso et l'observation du même pixel i. Les variables cachées X_i^1 et X_i^2 peuvent prendre des valeurs de l'ensemble $\mathcal{L} = \{0, 1\}$, où 0 correspond au fond et 1 correspond au texte.

Pour les relations entre les variables observées et les variables cachées, c'est à dire le processus de dégradation, nous supposons un MRF de premier ordre, ce qui signifie que nous supposons les deux conditions suivantes (une hypothèse habituelle dans le cadre MRF-MAP — voir le chapitre 3, section 3.2.3) :

1. Les observations Y_i sont indépendantes sachant les variables cachées X_i^1 et X_i^2.
2. $P(Y_i|X^1, X^2) = P(Y_i|X_i^1, X_i^2)$

En conséquence, le graphe de dépendances est structuré comme il este montré dans la figure 5.1b. Il contient les types de cliques suivants :

— les cliques d'ordre un et deux du sous graphe X^1 ;
— les cliques d'ordre un et deux du sous graphe X^2 ;
— les cliques *inter-champs* entre X^1, X^2 et Y contenant trois sommets, un de chaque champ. Les cliques inter-champs d'ordre deux ne contribueront pas à l'énergie, ç.à.d. leur potentiel sera mis à zéro.

La probabilité jointe de toutes les variables du graphe (cachées et observées) est donc donnée comme suit :

$$P(x^1, x^2, y) = \frac{1}{Z} \exp\big\{-\big(U(x^1) + U(x^2) + U(x^1, x^2, y)\big)\big\} \tag{5.1}$$

où Z est la fonction de partition définie par $\sum_{x^1,x^2} \exp\{-\big(U(x^1) + U(x^2) + U(x^1, x^2, y)\big)\}$. En séparant $\frac{1}{Z}$ en deux facteurs $\frac{1}{Z_1}$ et $\frac{1}{Z_2}$ et en fusionnant les cliques impliquant des variables cachées uniquement en une fonction unique $U(f^1, f^2)$, ce qui correspond à un simple changement de notation, nous obtenons :

$$P(x^1, x^2, y) \;= \frac{1}{Z_1} \exp\big\{-U(x^1, x^2)\big\} \frac{1}{Z_2} \exp\big\{-U(x^1, x^2, y)\big\} \tag{5.2}$$

En utilisant le théorème de Hammersely-Clifford [HC68, Bes74] et la règle de Bayes, nous pouvons interpréter (5.2) comme un problème Bayesien ce qui nous emmène à :

$$P(x^1, x^2, y) \ = P(x^1, x^2) P(y|x^1, x^2) \tag{5.3}$$

Le premier facteur correspond à la connaissance *a priori* et le deuxième facteur correspond à la vraissemblance des données. Regardons maintenant de plus près le *prior* $p(x^1, x^2)$. Dans la dérivation ci-dessus nous avons vu qu'il était composé de cliques impliquant des variables de deux champs cachés X^1 et X^2 seulement, et qu'il n'y avait aucune clique impliquant des variables des deux champs à la fois. Pour cette raison,

$$\begin{aligned} P(x^1, x^2) &= \frac{1}{Z} \quad \exp\left\{-U(x^1, x^2)\right\} \\ &= \frac{1}{Z} \quad \exp\left\{-\left(U(x^1) + U(x^2)\right)\right\} \\ &= \frac{1}{Z_{1'}} \quad \exp\left\{-U(x^1)\right\} \cdot \\ &\quad \frac{1}{Z_{2'}} \quad \exp\left\{-U(x^2)\right\} \\ &= P(x^1) P(x^2) \end{aligned} \tag{5.4}$$

En d'autres termes, l'écriture sur le recto est indépendante de l'écriture sur la page verso, ce qui prend tout son sens puisque les deux pages ne sont pas nécessairement influencées l'une par l'autre — elles peuvent même avoir été créés par des auteurs différents. Toutefois, cette indépendance concerne uniquement la situation dans laquelle aucune observation n'a été faite. En présence d'observations (l'image numérisée), les deux champs cachés ne sont pas indépendants en raison des cliques de trois sommets impliquant une paire de variables cachées et une variable observée. Intuitivement parlant, cela peut être illustré par l'exemple suivant : si l'observation d'un pixel donné suggère qu'au moins un des deux côtés du document contient du texte à cet endroit (si, par exemple, le niveau de gris du pixel est plutôt faible pour un document à texte noir sur fond blanc), alors la connaissance que l'étiquette recto est égale à *fond* va augmenter la probabilité que le pixel verso sera *texte*.

Les termes $U(x^1)$ and $U(x^2)$ correspondent à deux modèles de type Potts [Pot52, Li01], un pour chaque champ :

$$\begin{aligned} U(x^1) + U(x^2) \ = \ U(x^1, x^2) \ &= \ \sum_{\{i\} \in \mathcal{C}_1} \alpha^1 x_i^1 + \sum_{\{i,i'\} \in \mathcal{C}_2} \beta^1_{i,i'} \delta_{x_i^1, x_{i'}^1} \\ &+ \ \sum_{\{i\} \in \mathcal{C}_1} \alpha^2 x_i^2 + \sum_{\{i,i'\} \in \mathcal{C}_2} \beta^2_{i,i'} \delta_{x_i^2, x_{i'}^2} \end{aligned} \tag{5.5}$$

où $\mathcal{C}_1$ est l'ensemble de cliques à un sommet unique, $\mathcal{C}_2$ est l'ensemble de cliques de taille 2 (paires) et δ est le delta de Kronecker (voir l'équation (3.17) dans le chapitre 3, page 48). Les paramètres $\alpha^1, \alpha^2, \beta^1$ et β^2 servent à gérer la connaissance *a priori* sur l'équilibre entre les étiquettes *texte* et *fond* ainsi que sur les forces d'interaction entre les pixels voisins.

Le terme $U(x^1, x^2, y)$ est relié à la vraisemblance des données qui est dérivée à partir du modèle de dégradation. Nous proposons un modèle qui permet de tenir compte de plusieurs variations et dégradations :

— Il n'y a aucune restriction sur le niveau de gris ou sur la couleur du texte recto ou du texte au verso. Par contre, nous partons de l'hypothèse d'une variabilité constante ou Gaussienne des couleurs. Cela peut être étendu à des mélange de distributions Gaussienne (GMM) sans problème ;
— Une atténuation éventuelle linéaire des couleurs du verso est prise en compte ;
— L'encre sera supposé 100% opaque, c.à.d. qu'un pixel du texte recto recouvre totalement un pixel verso correspondant. Cela ne s'applique pas aux pixels du recto qui font partie du fond ;
— Du bruit Gaussien supplémentaire venant du matériel d'acquisition est pris en compte.

L'hypothèse d'opacité de l'encre pourrait théoriquement conduire à des problèmes dans une situation où les traits du verso sont plus sombres que les traits du recto. Toutefois, cette situation est assez peu probable en raison du fait que les traits du verso sont normalement éclaircis et désaturés par le processus de dégradation, plus particulièrement la bavure de l'encre. Nous tenons à clarifier que l'hypothèse donnée n'a pas été choisie pour simplifier le problème. Au contraire, l'hypothèse d'une encre partiellement transparente nous aurait permis d'inclure un modèle de mélange dans la vraisemblance, donc d'injecter une connaissance sur le pixel du verso même si le pixel recto est classé comme texte. D'ailleurs, cela ne remettrait pas en cause notre modèle, ni l'algorithme d'inférence associé.

A partir des hypothèses ci-dessus, le terme $U(x^1, x^2, y)$ pour la vraisemblance peut être donné comme suit :

$$U(x^1, x^2, y) = -\log p(y|x^1, x^2) = \sum_i -\log \mathcal{N}(y_i;\ \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \tag{5.6}$$

où $\boldsymbol{\mu}_i$ est la moyenne de la classe f_i et $\boldsymbol{\Sigma}_i$ est la matrice de covariance pour la classe f_i. $\boldsymbol{\mu}_i$ est donnée comme ci-dessous :

$$\boldsymbol{\mu}_i \quad = \quad \begin{cases} \boldsymbol{\mu}_r & \text{si } x_i^1 = \textit{texte} \\ \boldsymbol{\mu}_v & \text{si } x_i^1 = \textit{fond} \text{ et } x_i^2 = \textit{texte} \\ \boldsymbol{\mu}_{bg} & \text{sinon} \end{cases} \tag{5.7}$$

où $\boldsymbol{\mu}_r, \boldsymbol{\mu}_v$ et $\boldsymbol{\mu}_{bg}$ sont, respectivement, et *dans l'image dégradée*, la moyenne de la classe recto, de la classe verso, et de la classe fond. Les matrices de covariance sont définies de façon similaire.

5.4 Segmentation par un modèle hiérarchique

Le modèle développé dans cette section se focalise sur l'amélioration d'un modèle hiérarchique classique, l'arbre quaternaire de Markov [BS94, LPH00], dont le graphe de dépendances est illustré dans la figure 5.1d. L'objectif est de résoudre le problème du au manque d'invariance par rapport à la translation, ç.à.d. la non stationnarité du modèle, tout en gardant ses bonnes propriétés de modélisation hiérarchique. Notre modèle proposé combine donc plusieurs avantages :

— il est capable de s'adapter aux différentes caractéristiques de l'image avec une structure de graphe hiérarchique, semblable à l'arbre quaternaire ;
— le processus aléatoire au niveau de la base, ç.à.d. là où chaque site correspond à un pixel de l'image d'entrée, est stationnaire, donc invariant aux translations de l'image d'entrée ;
— il permet une minimisation d'énergie efficace par *graph cuts* pour une classe très intéressante de fonctions d'énergie.

Dans cette section nous discuterons des modèles sous forme de réseaux Bayesiens, donc des graphes orientés sans cycles, en tenant compte du sens des arêtes. Dans ce qui suit, nous noterons par i^- l'ensemble de parents du sommet i et par i_- l'ensemble de descendants. La nature hiérarchique du modèle induit un partitionnement de l'ensemble V de sommets du graphe $\mathcal{G} = \{V, E\}$ en sous ensembles disjoints $V^{(l)}$ correspondant aux différents niveaux l, ç.à.d. $V = \bigcup_{l \in 0..L-1} V^{(l)}$. L'ensemble $V^{(0)}$ correspond au niveau de base, donc à l'image à segmenter.

A chaque sommet i est associé une variable discrète X_i prenant des valeurs dans l'ensemble $\mathcal{L} = \{0, \ldots, C-1\}$ où C est le nombre de classes. Nous dénoterons par X l'ensemble des variables du champ, et par $X_{V^{(l)}}$ le champ correspondant au niveau l. Dans le cas du modèle sous forme d'arbre quaternaire [BS94, LPH00], le graph est un arbre avec une racine unique $r = V^{(L-1)}$, quatre descendants par sommet et un seul parent par sommet, à l'exception de la racine.

A chaque variable cachée X_i est associée une variable observée Y_i. Comme pour le modèle à double couche, les indépendances conditionnelles habituelles s'appliquent — voir le chapitre 3, section 3.2.3. Le graphe de dépendances du modèle arbre quaternaire est illustré dans la figure 5.1d.

L'objectif est d'estimer les variables cachées X à partir des variables observées Y par l'estimateur MAP :

$$\widehat{x} = \arg\max_{x \in \Omega} p(x|y), \tag{5.8}$$

donc de minimiser l'énergie correspondant. L'absence de cycle dans le graphe de dépendances permet l'application d'une extension de l'algorithme de Viterbi [Vit67, LPH00] — voir aussi le chapitre 3, section 3.4.1.

L'inconvénient majeur de l'arbre quaternaire est le manque de stationnarité dans le processus aléatoire des feuilles, $V^{(0)}$. En effet, deux sommets du graphe correspondant à deux pixels voisins de l'image (le voisinage ici étant celui de l'image et non pas celui du graphe) peuvent partager, oui ou non, un parent commun dans le niveau au dessus. Cela dépend de la position de la paire sur la grille de l'image. En général, toutes les paires de pixels de l'image sont connectées par un chemin dans l'arbre, en ignorant l'orientation des arêtes. Cela s'applique bien évidemment particulièrement aux pixels voisins (dans l'image). Par contre, selon la position dans l'image, ce chemin peut être très court, par exemple en passant par un parent commun dans le niveau au dessus, ou très long, par exemple en passant par la racine.

Motivé par cet inconvénient, nous proposons une extension de la structure graphique que nous illustrons dans les figures 5.2a-d. Pour des raisons pédagogiques nous présentons le cas en une dimension, ç.à.d. un arbre dyadique. Le cas de départ, un arbre dyadique représentant l'arbre quaternaire du cas 2D, est montré dans la figure 5.2a. Dans un premier temps, un deuxième arbre dyadique est ajouté au graphe, connectant toutes les paires de voisins non-connectées par un parent direct. Dans le cas 2D, trois arbres quaternaires sont ajoutées. Le problème est maintenant résolu pour le premier niveau, où le nombre de parents pour chaque pixel est des lors de quatre (pour le cas 2D ; 2 parents pour le cas 1D illustré dans la figure). Le résultat de cette première opération est illustré dans la figure 5.2b. Cette opération est maintenant répétée pour les niveaux supérieurs. A chaque opération, des nouveaux arbres connectent des sites de l'arbre d'origine, mais aussi des sites venant de nouveaux arbres ajoutés lors des opérations précédentes. Le résultat final peut être vu dans la figure 5.2d. Le graphe final n'est plus une pyramide, puisque chaque niveau a le même nombre de sommets que le niveau précédent. En général, chaque sommet a quatre parents dans le cas 2D (2 dans le cas 1D affiché), à l'exceptions des sommets sur les bords et les sommets du

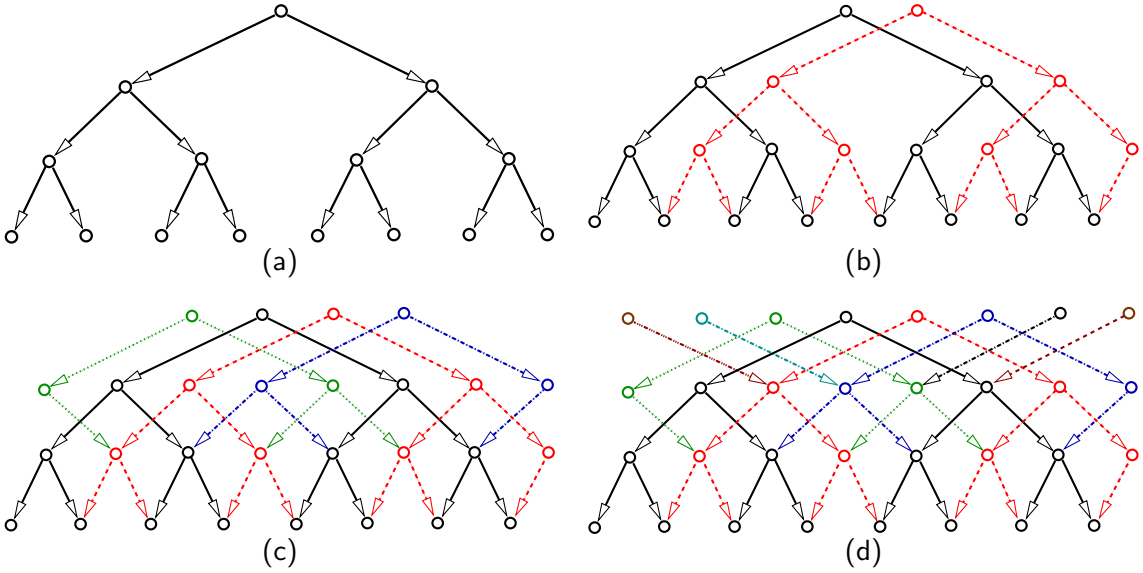


Figure 5.2 – Une représentation uni-dimensionnelle et pédagogique de l'extension, pas à pas, de l'arbre quaternaire (a) au modèle proposé sous forme de cube (d).

niveau le plus élevé remplaçant la racine.

Le graphe complet peut être implémenté de manière efficace par un cube de taille $N \times M \times \lceil \log_2 \max(N, M) \rceil$, où $N \times M$ est la taille de l'image d'entrée. En pratique, la hauteur complète n'est pas toujours nécessaire, bien qu'une certaine hauteur minimale est nécessaire pour assurer un minimum d'interactions spatiales entre les sommets. Dans cette représentation, la structure graphique est directement donnée par les coordonnées des sommets dans le cube, c.à.d. que les coordonnées des parents et des descendants d'un sommet i peuvent être calculés directement.

Le graphe décrit ci-dessus et montré dans la figure 5.2d (en une version 1D) correspond aux sites des variables cachées du problème. Il est complété par des variables observées afin de servir à la modélisation Bayesienne. Comme pour l'arbre quaternaire, nous ajoutons une variable observée à chaque variable cachée, et nous appliquons les conditions d'indépendance conditionnelle des observations. Les paramètres du modèle sont donc complètement définis par trois types de distributions :

— la distribution *a priori* du niveau le plus élevé du cube : $p(x_i), i \in V^{(L-1)}$;
— les probabilités de transition : $p(x_i|x_{i^-})$;
— la vraisemblance des observations : $p(y_i|x_i)$.

Le modèle complet avec les variables cachées est montré dans la figure 5.3.

Le modèle étant de type génératif, on s'intéressera naturellement à l'échantillonnage de nouvelles observations. Le modèle étant causal et sans cycles (en tenant compte du sens des arêtes), cela peut être fait par une passe du haut vers le bas afin d'échantillonner les variables cachées, et ensuite par un échantillonnage indépendant pour chaque variable observée.

Les algorithmes d'inférence demanderont des observations à chaque niveau du modèle. Or, elles sont disponibles de manière directe seulement pour l'image d'entrée, ç.à.d. pour le niveau le plus bas. Les autres observations peuvent être obtenues par un processus de ré-échantillonnage, en tenant compte de la la structure du graphe.

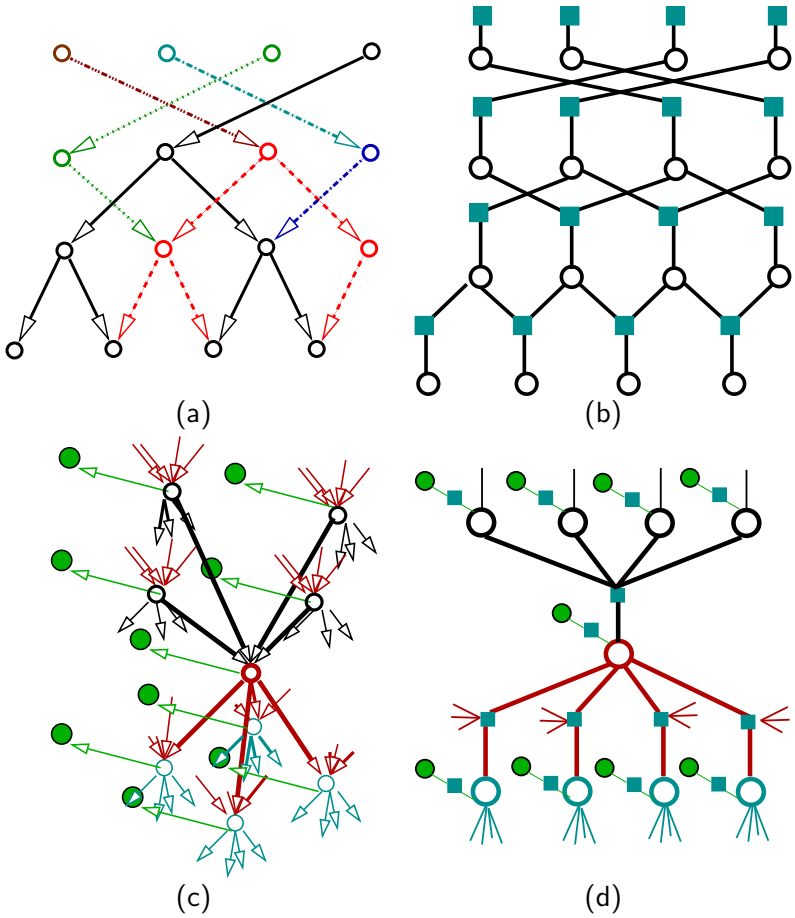


Figure 5.3 – Le modèle de cube en version simplifiée 1D (en haut) et sous sa forme complète en 2D pour un sommet et ses parents et descendants directes (en bas). Le graphe de dépendances est affiché à gauche, un graphe de facteurs montrant la décomposition de l'énergie à droite.

5.5 Minimisation d'énergies

Les structures graphiques différentes des deux modèles, le MRF à double couche de la section 5.3 et le cube de la section 5.4, suggèrent des approches de minimisation différentes :

- le MRF a double couche est caractérisé par une faible connexité et un grand nombre de termes sous-modulaires. Nous proposons une adaptation de la méthode par coupure minimale dans un graphe appliquée au graphe de dépendance de notre modèle. Nous avons étendu l'algorithme *mouvement d'expansion-*α proposé par Boykov et al. [BK04] et amélioré par Kolmogorov et al. [KZ04] ;
- la stratégie de minimisation du modèle de cube dépendra de la forme spécifique de la distribution des transitions. Pour une forme générale, nous proposons la propagation de croyances comme approche. Par contre, nous montrerons qu'une stratégie très efficace à base de *graph cuts* peut être trouvée pour une classe très intéressante de distributions de transitions ;

Les sections suivantes donneront quelques indications sur ces méthodes.

5.5.1 Séparation recto/verso d'images de document

Notre modélisation par MRF à double couche a transformé le problème de segmentation multi-classes (*fond, recto, verso, recto+verso*) en un problème binaire (*texte, fond*) sur une quantité plus élevée de variables cachées. Cette stratégie s'avère avantageuse d'un point de vu optimisation, rendant *presque* inutile l'application d'un algorithme itératif de type *mouvement d'expansion-*α.

Le seul obstacle reste la condition de sous-modularité tellement importante pour les algorithmes de type *graph cuts*. Comme on peut constater, cette condition est naturellement satisfaite pour les potentiels d'énergie des cliques du modèle de Potts. Par contre, les potentiels d'énergie log-Gaussiens issus de la vraisemblance des données (5.6) ne sont pas nécessairement sous-modulaires. En effet, la sous-modularité de ces fonctions dépend de l'observation Y_i sur chaque site. L'énergie de ce type de clique va donc être sous-modulaire pour certains pixels et non sous-modulaire pour d'autres.

Pour pallier à ce problème nous avons conçu un algorithme itératif alternant entre une estimation des étiquettes cachées du champ recto X^1 en figeant les étiquettes X^2 du champ verso et vice versa. Les étiquettes figées étant constantes, les fonctions d'énergie correspondantes sont désormais unaires, ç.à.d. dépendant d'une seule variable cachée, et donc naturellement sous-modulaires. Nous pouvons toutefois améliorer l'algorithme en figeant seulement les variables cachées pour les pixels dont les termes de vraisemblance le demandent, ç.à.d. pour les pixels dont la fonction d'énergie n'est pas sous-modulaire. Pour les autres, les deux variables cachées, recto et verso, peuvent être estimées en même temps et de façon optimale.

La construction du graphe s–t pour notre problème est basée sur une décomposition de la fonction d'énergie globale (5.1) selon les conditions de sous-modularité pour les pixels de l'image d'entrée. Tout d'abord nous simplifierons l'écriture de l'équation (5.1) en l'exprimant comme une somme de potentiels unaires U_1 et de deux types de potentiels par paires U_2 et U_2' dont seulement les termes U_2' sont potentiellement non sous-modulaires :

$$\begin{aligned} U(x^1, x^2, y) &= \sum_{\{i\}\in\mathcal{C}_1} \left[\alpha^1 U_1(x_i^1) + \alpha^2 U_1(x_i^2)\right] \\ &+ \sum_{\{i,i'\}\in\mathcal{C}_2} \left[\beta_{i,i'}^1 U_2(x_i^1, x_{i'}^1) + \beta_{i,i'}^2 U_2(x_i^2, x_{i'}^2)\right] \\ &+ \sum_{\{i\}\in\mathcal{C}_1} U_2'(x_i^1, x_i^2; y_i) \end{aligned} \tag{5.9}$$

Ensuite, une matrice H est calculée indiquant pour chaque pixel i s'il est, oui ou non, sous-modulaire :

$$H_i = \begin{cases} 1 & \text{si} \quad U_2'(0,0,y_i) + U_2'(1,1,y_i) \leq U_2'(0,1,y_i) + U_2'(1,0,y_i) \\ 0 & \text{sinon} \end{cases} \tag{5.10}$$

Pour le cas où un sous-ensemble du champ X^1 est figée, et le champ complet X^2 et le sous-ensemble complémentaire du champ X^1 sont estimés de manière optimale, les termes de l'énergie

peuvent être séparés de manière suivante :

$$
\begin{aligned}
U^{\mapsto 2}(x^1, x^2, y, H) \;=\; & \sum_{\{s\}\in\mathcal{C}_1:H_i=0} \alpha^1 U_1(x_i^1) \\
+ & \sum_{\{i\}\in\mathcal{C}_1:H_i=1} \alpha^1 U_1(x_i^1) \\
+ & \sum_{\{i\}\in\mathcal{C}_1} \alpha^2 U_1(x_i^2) \\
+ & \sum_{\{i,i'\}\in\mathcal{C}_2:H_i=0\wedge H_{i'}=0} \beta^1_{s,s'} U_2(x_i^1, x_{i'}^1) \\
+ & \sum_{\{i,i'\}\in\mathcal{C}_2:H_i=1\wedge H_{i'}=1} \beta^1_{s,s'} U_2(x_i^1, x_{i'}^1) \\
+ & \sum_{\{i,i'\}\in\mathcal{C}_2:H_i\neq H_{i'}} \beta^1_{s,s'} U_2(x_i^1, x_{i'}^1) \\
+ & \sum_{\{i,i'\}\in\mathcal{C}_2} \beta^2_{s,s'} U_2(x_i^2, x_{i'}^2) \\
+ & \sum_{\{i\}\in\mathcal{C}_1:H_i=0} U_2'(x_i^1, x_i^2; y_i) \\
+ & \sum_{\{i\}\in\mathcal{C}_1:H_i=1} U_2'(x_i^1, x_i^2; y_i)
\end{aligned}
\tag{5.11}
$$

A partir de l'équation (5.11), qui est sous-modulaire, il est facile de construire un graphe $s{-}t$ avec la méthode de Kolmogorov et Zabih [KZ04] — voir aussi le chapitre 3, section 3.4.4. Le cas contraire, où une partie du champ X^2 est figée, suit un raisonnement similaire.

5.5.2 Segmentation par un modèle hiérarchique

La minimisation de la fonction d'énergie du modèle sous forme de cube est difficile dans la version générale, ç.à.d. quelque soit la définition des distributions de probabilité de transition. Sans contrainte supplémentaire, l'énergie peut être minimisée par propagation de croyances, une technique approximative conçue pour des graphes de structure générale [Pea88] — voir aussi le chapitre 3, section 3.4.2.

La figure 5.3 montre le graphe de dépendances du modèle ainsi qu'un graphe de facteurs. Ce dernier met en évidence la manière par la laquelle la probabilité jointe se factorise : les sommets ronds correspondent aux variables du modèle, les sommets carrés correspondent aux facteurs de la probabilité jointe, donc au termes de l'énergie. La propagation de croyances procède en passant des messages entre les sommets de ce graphe, chaque message étant une fonction des variables associées aux sommets et à certains voisins. Dans notre cas concret, l'algorithme alternera entre des étapes du haut ver le bas du cube, et des étapes du bas vers le haut.

Interprétation des variables cachées des niveaux supérieurs

Nous proposons une méthodologie pour l'interprétation des variables cachées x_i des niveaux supérieurs et donc des distributions conditionnelles $p(x_s|x_{s^-})$, dite *distributions de transitions*. En tenant compte des invariances statistiques d'un corpus d'images similaires, on déduit que :

1. le modèle d'indépendance de la structure graphique est satisfait. Une variable x_i doit être indépendante de toutes les variables ayant un niveau inférieur sachant ses parents x_i^- ;
2. les probabilités conditionnelles sont différentes des probabilités obtenues sur des images aléatoires ;
3. les probabilités conditionnelles sont proches pour les images du même corpus.

Pour des raisons de simplicité, nous considérons le cas de la segmentation en 2 classes ($C = 2$). Soit i un sommet du cube et soit l son niveau. Nous appellerons U_i l'ensemble des sommets du niveau 0 qu'on peut atteindre par un chemin orienté et direct à partir de i. U_i est une région carrée de l'image de taille $2^l * 2^l$. De manière naturelle on peut définir la classe de x_i comme étant la classe ayant le nombre maximal de variables dans x_{U_i}. Cette interprétation suggère deux stratégies pour la définition des probabilités de transitions :

— Une définition non-paramétrique par tableaux ;
— Le fitting d'une fonction paramétrique sur des données d'apprentissage. Cette stratégie a été choisie dans la prochaine section.

Minimisation par graph cuts

Dans le cas du modèle proposé, tous les termes d'énergie ne sont pas sous-modulaires, en particulier les termes correspondant au logarithme des probabilités de transition $\ln p(x_s|x_{s^-})$. Le modèle général ne peut donc pas être résolu de manière exacte par *graph cuts*. Cependant, pour une large classe de distributions, ayant des propriétés très intéressantes, une solution exacte par *graphe cuts* peut être trouvée pour le cas binaire, c.à.d. deux étiquettes possible par variable cachée. En tenant compte de l'interprétation des variables décrite ci-dessus (section 5.5.2), nous proposons un terme de régularisation basé sur le nombre d'étiquettes parentales qui sont égales à l'étiquette du descendant en question :

$$p(x_i|x_{i^-}) = \frac{1}{Z}\alpha_l^{\xi(x_i,x_{i^-})} \tag{5.12}$$

où α_l est un paramètre dépendant de l'échelle l, $\xi(x_i, x_{i^-})$ est le nombre d'étiquettes dans x_{i^-} égales à x_i et Z est une constante de normalisation. Cette distribution favorisera des régions homogènes avec une force d'interaction qui dépend de l'échelle. Cela correspond clairement aux objectifs de la méthode.

La fonction peut être décomposée en une somme de termes par paires :

$$\ln p(x_i|x_{i^-}) = \sum_{i' \in i^-} [(\ln \alpha)\ \delta_{x_i,x_{i'}}] - Z \tag{5.13}$$

où $\delta_{a,b}$ est le delta de Kronecker (voir équation (3.17) dans le chapitre 3, page 48). Nous voyons que chaque terme de cette somme est sous-modulaire si $\alpha \leq 1$, ce qui correspond aux configurations intéressantes favorisant des régions homogènes. Un graphe $s{-}t$ peut donc être trouvé facilement avec la méthode Kolmogorov et Zabih [KZ04] — voir aussi le chapitre 3, section 3.4.4. La figure 5.4 montre un graphe $s{-}t$ construit pour le graphe de dépendances montré dans la figure 5.3c dans le cas ou $C{=}2$. Des problèmes de type multi-étiquettes $(C{>}2)$ peuvent être traités par l'algorithme *movuement d'expansion-*α avec un graphe $s{-}t$ similaire — voir aussi le chapitre 3, section 3.4.4.

La complexity de calcul dépend de la méthode de minimisation utilisée.

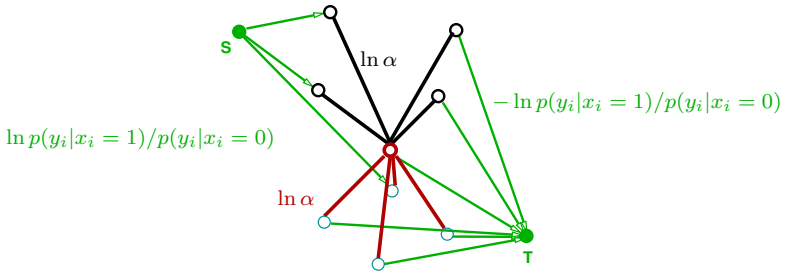


Figure 5.4 – Un extrait du graphe $s{-}t$ construit pour la version binaire du problème.

Propagation de croyances La minimisation est de complexité de calcul de $O(I{\cdot}N{\cdot}M{\cdot}(H{-}1){\cdot}C^5)$ où I est le nombre d'itérations, N et M sont les dimensions de l'image d'entrée, et H est la hauteur du cube, qui est bornée par $\lceil \log_2 \max(N, M) \rceil$. La complexité en espace mémoire est donnée par $N \cdot M \cdot (H-1) \cdot 15C$ par variable. En pratique, la propagation de croyances est applicable pour des nombres de classes C assez petits, ç.à.d. 2, 3 ou 4 au maximum. Cela suffit pour un grand nombre de problèmes.

Graph cuts L'inférence par *graph cuts* est bien moins coûteuse. La complexité est donnée par $O(E * f)$, où E est le nombre d'arêtes dans le graphe $s{-}t$ est F et le flux maximal. Nous utilisons l'algorithme Ford-Fulkerson dans la version implémentée par Boykov et Kolmogorov [BK04]. Cette version a été optimisée pour des structures de graphes typiques en vision. Sa complexité est proche de la linéarité en pratique [BVZ01].

5.6 Estimation de paramètres

Les fonctions d'énergie associées aux deux modèles ont été définies aux valeurs des paramètres près. L'identification des ces paramètres est donc nécessaire avant de pouvoir appliquer les modèles. Nous estimerons les paramètres sans données d'apprentissage directement à partir des images d'entrée. Comme les réalisations du/des champ(s) cachés X (ou X^1 et X^2) ne sont pas disponibles, les paramètres du modèle sont estimés à partir des données observées. Les algorithmes non-supervisés classiques procèdent par des estimations itératives du champ caché, ou par la maximisation de son espérance (voir le chapitre 3, section 3.5.2). En pratique cela ne s'avère pas toujours nécessaire. Nous avons choisi d'estimer les paramètres de manière supervisée à partir d'une version lissée de l'image d'entrée.

Les paramètres sont estimés à partir de l'image entrée après un *clustering* de type k-moyennes suivi d'un post-traitement avec un lissage robuste de type filtre médian. Pour une application au problème de séparation recto/verso, une segmentation en 3 classes est ciblée. Les étiquettes obtenues par la méthode k-moyennes étant non ordonnées et non pourvues de signification, la reconnaissance de leur type (*recto, verso, fond*) s'avère nécessaire. Nous avons conçu un algorithme capable de détecter ce type sans nécessité de recourir aux informations provenant de l'histogramme de l'image, ç.à.d. sans connaissance *a priori* sur la "couleur" (le niveau de gris) du texte. La méthode se sert de statistiques sur les changements d'étiquettes entre pixels voisins et d'une analyse de composantes connexes. En supposant que les composantes connexes verso ont tendance à être couvertes, et donc d'être séparées en plusieurs parties, la signification des étiquettes peut être déterminée.

Les paramètres du MRF

Les paramètres du modèle de Potts $(\alpha^1, \alpha^2, \beta^1, \beta^2)$, voir l'équation (5.5), sont estimés avec la méthode par moindres carrés proposée par Derin et al. [DE87], voir aussi le chapitre 3, section 3.5.1. Le champ recto étant plus stable, nous estimons les paramètres sur ce champ uniquement, car toutes ses étiquettes sont directement liées à l'image observée. Les paramètres du champ verso sont calculés à partir des paramètres du champ recto en supposant que, statistiquement parlant, le champ verso est une version inversée du champ recto.

Les paramètres du cube

Les probabilités a priori des variables cachées du niveau le plus haut peuvent être estimées en utilisant des techniques d'histogramme. Les paramètres α_l des distributions conditionnelles peuvent êtres estimés de manière similaire aux paramètres du double MRF. Au lieu de mettre en relation chaque pixel avec son voisinage, on peut mettre en relation chaque pixel avec ses parents :

$$\frac{P(X_i{=}x_i|X_{i^-}{=}x_{i^-})}{P(X_{i'}{=}x_{i'}|X_{i'^-}{=}x_{i^-})} = \frac{\alpha_l^{\xi(x_i,x_{i^-})}}{\alpha_l^{\xi(x_{i'},x_{i^-})}} \tag{5.14}$$

Comme pour [DE87], on cherche des paires de pixels ayant des étiquettes différents x_i et x'_i tel que leur parents partagent les mêmes étiquettes $x_{i'}$. En remplaçant les probabilités conditionnelles par des probabilités absolues et en prenant le logarithme de l'expression, un système linéaire surdéterminé est obtenu. Ce dernier peut être résolu par des techniques qu'on peut résoudre avec des techniques de moindres carrés.

Les paramètres du modèle d'observation

Le modèle d'observation est le même pour les modèles, le MRF à double couche et le cube de Markov. Comme mentionné auparavant, il s'agit d'un modèle Gaussien par pixel. Grâce à l'indépendance des observations sachant les variables cachées, les paramètres des modèles d'observation sont estimés à l'aide des estimateurs classiques de type vraisemblance maximale (la moyenne empirique et les covariances empiriques).

5.7 Résultats sur les images de documents

Les modèles ayant été conçus avec des objectifs différents et ciblant des types de documents différents, une comparaison directe et quantitative sur un corpus unique s'avère assez difficile. De plus, leur conception s'étalent sur plusieurs années. Malgré cette difficulté nous avons essayé de compléter les évaluations individuelles par une comparaison partielle. Ici nous donnerons des résultats sur une seule base d'images, nous renvoyons le lecteurs vers [Wol08, Wol10, WG10, Wol06] pour des expériences supplémentaires sur d'autres bases.

Nous avons choisi un ensemble de données composé de 104 pages de texte de faible qualité imprimées à partir du 18^e siècle, issues d'une numérisation des gazettes de Leyde. Ce journal en langue française a été imprimé de 1679 à 1798 aux Pays-Bas afin d'échapper à la censure en France au 18^e siècle et concerne des nouvelles du monde. Les bulletins sont actuellement utilisés

	Base A (104 imgs) Moyenne résolution Segment. 3 classes		Base B (9 imgs) Haute résolution Segment. 3 classes	
Méthode	Rappel	Précision	Rappel	Précision
Sans restauration	65.65	49.91	$\ll$	$\ll$
Niblack [Nib86] Ⓢ	$\ll$	$\ll$	–	–
Sauvola et al. [SSHP97] Ⓢ	78.75	66.78	–	–
K-moyennes (k=3)	78.57	69.43	61.23	51.74
Tonazzini et al. [TSB07]	‡ 41.00	‡ 30.05	–	–
Tonazzini et al. [TB04]	$\ll$	$\ll$	–	–
Tonazzini et al. [TB04] - toutes les 3 sources	‡ 50.52	‡ 33.90	13.13	25.43
MRF - Potts & α-exp. [KZ04]	81.99	72.12	69.10	58.42
MRF 4×4, recuit simulé	–	–	–	–
MRF double couche	83.23	74.85	75.76	68.08
Cube & graph cut	–	–	69.34	61.19

Ⓢ La méthode permet une segmentation en 2 classes seulement ; le verso n'est donc pas reconnu.
$\ll$ résultat de très faible qualité : une évaluation correcte est impossible
‡ résultat de très faible qualité : l'évaluation a été obtenue sur un sous-ensemble

Table 5.1 – L'évaluation des différentes méthodes de restauration appliquées aux images de documents scannés. **Les méthodes ne sont pas comparables sur des corpus différents !**

par plusieurs projets de recherche en sciences sociales et politiques, dont certains sont actuellement en collaboration avec notre équipe dans le cadre de projets de numérisation.

D'un point de vue traitement d'images, la difficulté des données se situe entre les manuscrits, considérés comme difficile, et les documents modernes imprimés, considérés comme plutôt faciles. Les images de tailles 1030×1550 pixels sont de qualité très faible par rapport aux textes imprimés modernes. La reconnaissance automatique des caractères est possible ; elle donne toutefois des résultats très moyens. La base comprend 104 images de moyenne résolution (base A, à 300 dpi) et 9 images de haute résolution (base B, 600 dpi).

Nous avons évalué toutes les méthodes sur un problème de restauration, la métrique étant la performance d'un logiciel de reconnaissance de caractères, à savoir Google Tesseract, un logiciel libre [5].

5.7.1 Évaluation quantitative et comparative par OCR

Les méthodes présentées dans ce mémoire ont été comparées à des méthodes différentes que nous pouvons classées en trois groupes :

— Des méthodes classiques de segmentation ou de binarisation de documents : le clustering par k-moyennes ; la binarisation de Niblack [Nib86] et la binarisation de Sauvola et al. [SSHP97] ;

5. `http://code.google.com/p/tesseract-ocr`

— Un modèle graphique probabiliste classique, à savoir un MRF classique avec un modèle de Potts et optimisation par *graph cuts* [KZ04] ;
— deux méthodes de séparation recto/verso bien connues [6] basées sur la séparation de sources par analyse de composantes indépendantes [TB04, TSB07].

Pour toutes les méthodes de restauration capables d'identifier les pixels de type *verso*, ces pixels ont été remplacés par la valeur moyenne du niveau de gris des pixels autour classé comme *fond*.

La table 5.1 donne les résultats sur les différentes méthodes, et sur des images différentes. Comme mentionné auparavant, les résultats ne sont pas comparables sur les différentes bases d'images. Les mesures données sont le rappel (le nombre de caractères correctement reconnus par rapport au nombre de caractères dans la vérité de terrain) et la précision (le nombre de caractères correctement reconnus par rapport au nombre de caractères sortis par l'OCR).

La grande surprise est la faible performance des méthodes basées sur la séparation de sources. Les résultats des ces méthodes n'ont pas permis de passer un OCR sur l'ensemble complet des pages. Les méthodes basées sur les modèles graphiques probabilistes arrivent à surpasser toutes les autres méthodes. Sans surprise, le MRF à double couche (voir section 5.3) donne les meilleurs résultats — il a été conçu pour cette tâche. On peut aussi remarquer que le cube surpasse le MRF plat (le modèle de Potts) sur les images de haute résolution. Pour la séparation recto/verso, un cube à double couche pourrait éventuellement donner encore des meilleurs résultats.

5.7.2 Séparation recto/verso d'images de documents

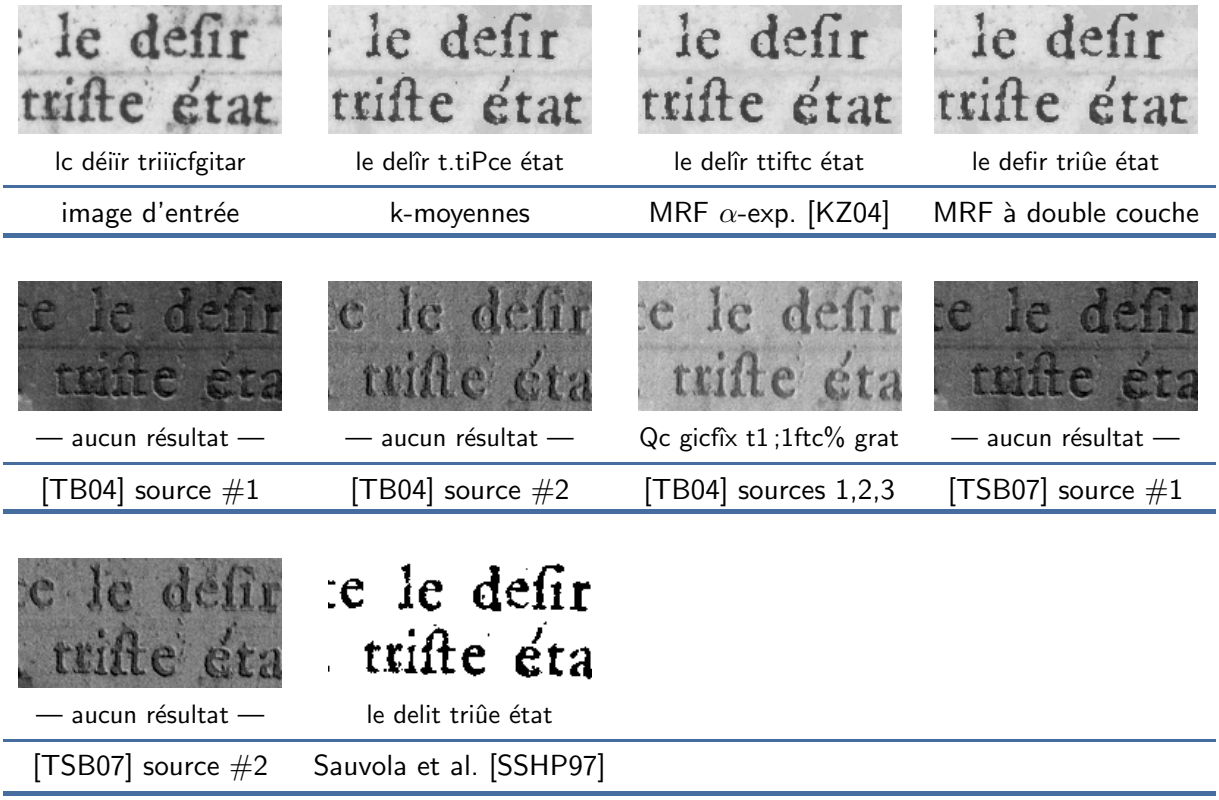
La figure 5.5 montre une partie des résultats de segmentation sur des images de moyenne résolution (base A). Le résultat du clustering par k-moyennes est très bruité comparé aux résultats obtenus par les modèles graphiques probabilistes. Le MRF à double couche améliore le résultat du MRF simple, ce qui se voit aussi dans la sortie de l'OCR.

Les résultats obtenus par les algorithmes Niblack [Nib86] et Sauvola et al. [SSHP97] confirment les faiblesses bien connues : Niblack produit des composantes fantômes, tandis que Sauvola et al. à tendance à couper des caractères du aux hypothèses sur la distribution des niveaux de gris, pas toujours satisfaites. Les deux algorithmes de séparation de sources ne sont pas automatique : parmi les composantes des résultats obtenus, l'utilisateur doit identifier celle correspondant à la composante du recto du document. Tous les résultats ont été traités avec les étapes de post-traitement recommandées par les auteurs en se basant sur les codes sources fournis par ces derniers.

5.7.3 Segmentation par un modèle hiérarchique

Le cube a été évalué de manière similaire au MRF à double couche : la base B contenant des images de haute résolution a servi comme corpus de test. La figure 5.6 montre des extraits avec les sorties OCR. Nous pouvons remarquer que les bords des caractères produits par le cube sont plus lissés que les bords produits par les autres méthodes. Comme pour le test sur les images de moyenne résolution, les résultat sont plutôt faibles pour les méthodes basées sur la séparation de sources. La figure 5.7 montre un extrait très agrandi d'un résultat comparant le MRF simple et le cube. La nature hiérarchique du cube lui permet de surpasser le MRF en réduisant les artefacts et en remplissant des trous dans la segmentation.

6. Nous remercions Prof. Anna Tonazzini de nous avoir fourni le code source de ses méthodes, pour son aide avec nos expériences et pour les discussions très intéressantes sur ce sujet.



lc déïïr triïïcfgitar	le delîr t.tiPce état	le delîr ttiftc état	le defir triûe état
image d'entrée	k-moyennes	MRF α-exp. [KZ04]	MRF à double couche
— aucun résultat —	— aucun résultat —	Qc gicfîx t1 ;1ftc% grat	— aucun résultat —
[TB04] source #1	[TB04] source #2	[TB04] sources 1,2,3	[TSB07] source #1
— aucun résultat —	le delit triûe état		
[TSB07] source #2	Sauvola et al. [SSHP97]		

Figure 5.5 – Résultats de restauration et de reconnaissance de caractères. Dans chaque ligne, de haut vers le bas : image résultat ; résultat de reconnaissance automatique de texte (OCR) ; méthode utilisée.

L'évaluation est complétée par une expérience sur les images synthétiques : 60 images de taille 512×512 ont été créées de manière synthétique et traitées avec des dégradations très importantes : filtrage passe bas, amplification de bandes de fréquences sous forme d'anneau causant des artefacts, codage de JPEG de basse qualité, bruit Gaussien supplémentaire avec des variances entre σ=20 et σ=40.

Le modèle a été comparé à plusieurs autres méthodes : MRF simple (modèle de Potts) et *graph cuts* [KZ04], l'arbre quaternaire [LPH00], clustering par k-moyennes. L'algorithme de k-moyennes est le seul dont la performance est améliorée si un lissage est appliqué avant le traitement. La table 5.2 donne les erreurs sur cette base.

5.8 Soustraction de fond dans la vidéo

Dans cette section nous compléterons l'ensemble de modèles proposés dans ce chapitre par un autre modèle conçu pour la segmentation de séquences vidéo, une tâche connue sous le nom de soustraction de fond. Ce traitement est une étape importante dans un grand nombre d'applications, comme le suivi de personnes, la reconnaissance de comportements etc.

La problématique est similaire à celle présentée dans la première partie de ce chapitre : il

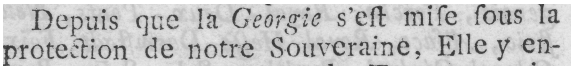
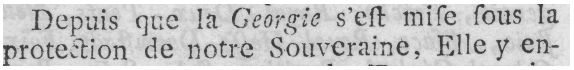
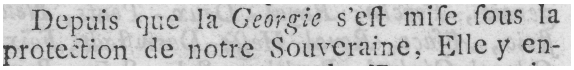
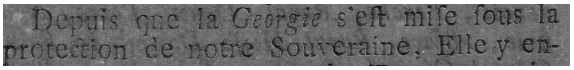
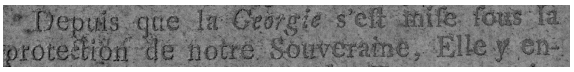
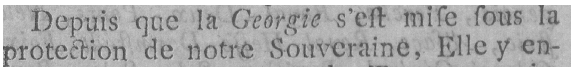
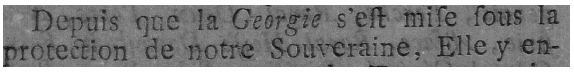
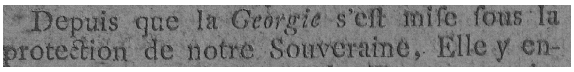
 Depuis que la Georgie s'c■'c mifc fous la protcîtion de notrè Souveraine, Ella-ry cn-	 Depuis que la Georgia s'e■ 'c mife fous la p·rote■ :'tion de notre Souveraine, Elle y en-
k-moyennes	MRF [KZ04]
 Depuis que la Georgia s'efi mife fous la proteèlion de notre Souveraine, Elle y en-	 — résultat OCR inutilisable —
cube+graph cuts	[TB04] source #1
 — résultat OCR inutilisable —	 — résultat OCR inutilisable —
[TB04] source #2	[TB04] source #3
 — résultat OCR inutilisable —	 — résultat OCR inutilisable —
[TB04] toutes les sources	[TSB07] source #1
 — résultat OCR inutilisable —	 — résultat OCR inutilisable —
[TSB07] source #2	image d'entrée

Figure 5.6 – Images restaurées et résultat d'OCR sur la base B. Dans chaque ligne, de haut vers le bas : image résultat ; résultat de reconnaissance automatique de texte (OCR) ; méthode utilisée.

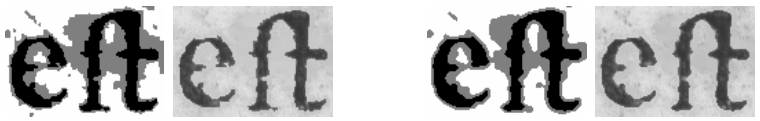


Figure 5.7 – Extrait très agrandi des résultats montrés dans la figure 5.6 (base B) : résultats de segmentation et de restauration pour le MRF [KZ04] (gauche) et le cube + *graph cuts* (droite).

Méthode	Erreur	Méthode	Erreur
K-moyennes	27.01	Cube-PC (4 niveaux, non-paramétrique)	6.82
K-moyennes (+ filtre passe bas)	9.01	Cube-PC (4 niveaux, paramétrique)	6.91
Quad arbre [LPH00]	7.57	Cube-PC (5 niveaux, paramétrique)	6.84
MRF-GC [KZ04]	6.28	Cube-GC (5 niveaux, paramétrique)	**5.58**

Table 5.2 – Erreurs de classification pour des méthodes différentes sur un problème de 2 classes (binarisation) et sur des images synthétiques ; GC=graph cuts, PC=propagation de croyance.

s'agit d'un problème de binarisation, donc de segmentation en deux classes, à savoir le fond de la vidéo et les objets en mouvement. Comme pour les images, les niveaux de gris ou les couleurs des pixels interviendront dans un terme d'attache aux données pour guider la classification. La différence principale réside dans l'échelle du modèle. Pour une image, soit un modèle global est envisageable pour décrire les distributions des observations, soit ou un modèle local à fenêtres larges est possible. Par contre, la nature générale des vidéos de type scènes naturelles, intérieur ou extérieur, nécessite d'établir un modèle de couleur pour chaque pixel. Les méthodes les plus simples — et les plus rapides — modélisent le fond par une distribution unimodale calculée par un filtre median [CGPP03], un filtre de Kalman [WADP97] ou des techniques similaires. Les techniques plus complexes passent par des distributions multi-modales comme les mélanges de distributions Gaussiennes (GMM) [PS06, SG00, ZvdH06]. Nous renvoyons le lecteur intéressé à la revue de littérature excellente par Park et al. [PF08].

Les algorithmes de type GMM sont devenus plus ou moins le standard pour la soustraction de fond dans les séquences vidéo, principalement en raison de leur capacité à modéliser plusieurs distributions de fond. Cela leur permet de gérer les scènes complexes, y compris des arbres en mouvement, le déplacement de drapeaux avec le vent etc. Pour chaque pixel, un GMM modélise tous les pixels d'un passé proche, y compris les pixels du fond et des objets en mouvement. Toutefois, il n'est pas toujours facile d'identifier les distributions du mélange appartenant au fond, ce qui perturbe les résultats du processus d'étiquetage pour chaque pixel.

5.8.1 Modélisation et minimisation

Ici, nous nous attaquons à ce problème en prenant la décision d'étiquetage de manière globale pour tous les pixels de plusieurs images consécutives en minimisant une fonction d'énergie et en tenant compte des relations spatiales et temporelles [WJ10]. Les interactions temporelles nécessitent un calcul de flot optique approximatif que nous avons conçu spécifiquement pour ce problème.

Notre travail s'appuie sur l'algorithme Stauffer-Grimson, une des méthodes de soustraction de fond les plus utilisées [SG00]. Le modèle est basé sur un ensemble de K distributions Gaussiennes par pixel modélisant les niveaux de gris — une extension au cas de couleur est immédiate. Pour chaque nouvelle frame de la vidéo et pour chaque pixel, le niveau de gris y est comparé à toutes les distributions. La distribution ayant la plus grande probabilité est sélectionnée, si y est suffisamment proche, sinon une nouvelle distribution est créée. Les paramètres de la distribution sélectionnée (moyenne μ, écart-type σ, poids w) sont mises à jour en se servant d'un paramètre de pas d'apprentissage.

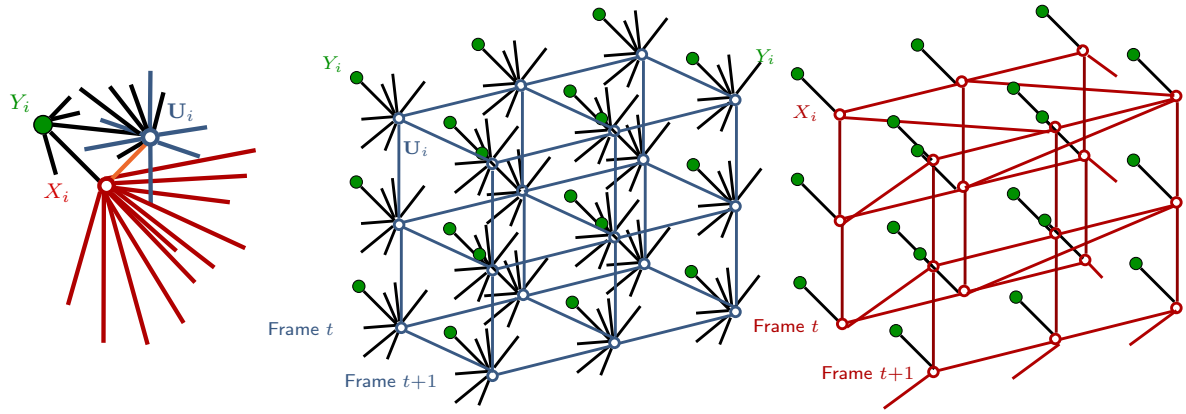


Figure 5.8 – Figure à consulter de préférence en couleur : (a) tentative d'esquisse du graphe de dépendances correspondant à l'équation complète pour un seul pixel (5.16) ; (b) le graphe pour l'estimation du champ $\boldsymbol{U}$ (3×3 pixels) ; (c) le graphe pour l'estimation du champ X.

La difficulté réside dans la décision si une gaussienne sélectionnée correspond à l'arrière-plan (AP) ou à un object en mouvement (OM). Dans les travaux d'origine [SG00], les distributions de chaque pixel sont triées par $\frac{w}{\sigma}$ et il est supposé que les 70% de la masse du mélange sont occupé par l'arrière-plan. Il va sans dire que cela n'est pas toujours satisfait, ce qui cause des erreurs de classification.

Nous avons amélioré la méthode en prenant l'intégralité de toutes les décisions d'un bloc spatio-temporel de manière globale en s'appuyant sur un modèle graphique. Le modèle comprend deux champs aléatoires, les deux étant indexés par les pixels du bloc spatio-temporel :

— un champ X modélisant les décisions de la segmentation (fond / forme) ;

— un champ U modélisant les vecteurs de mouvement.

Les deux champs sont inter-dépendants, l'un aidant à la régularisation de l'autre. Une méthode alternative est de créer un champ de Markov à état-mixte, où chaque variable associée à un pixel peut prendre soit une valeur discrète, soit une valeur continue. Dans [CBCFfY11], la variable aléatoire de chaque pixel peut prendre une seule valeure discrète correspondant à l'information « pixel décrivant un objet en mouvement », ou une valeur dans un interval contenu correspondant au niveau de gris du fond. L'avantage est la modélisation par un champ unique rendant (en théorie) l'inférence moins complexe. En pratique, les potentiels d'interactions dans [CBCFfY11] sont complexes, nécessitant une minimisation de l'énergie par l'algorithme glouton ICM, alors que notre méthode permet une minimisation exacte de chacun des champs X et U par *graph cuts*. Une autre différence est la modélisation du mouvement. Dans [CBCFfY11] le mouvement est booléen (un pixel est un mouvement, ou pas), alors que notre méthode passe par une estimation d'un champ de vecteurs de déplacement.

Dans notre méthode, les décisions individuelles pour les pixels i sont guidées par des mesures $\Delta(y_i)$ correspondant à la déviation du niveau de gris y_i de la distribution sélectionnée :

$$\Delta(y_i) = \frac{|y_i - \mu_i|}{\sigma_i} \tag{5.15}$$

La fonction d'énergie comprend un terme d'attache aux données impliquant $\Delta(y_i)$, ainsi qu'un modèle de Potts [Li01] (voir aussi la section 5.3) regularisant les interactions spatiales et temporelles,

les dernières étant guidées par un champ de mouvement $\boldsymbol{u}$:

$$\begin{aligned} E(x,y,\boldsymbol{u}) &= \alpha_d \sum_i E_d(y_i,x_i) \\ &+ \alpha_s \sum_{i\sim j} \delta(x_i,x_j) \\ &+ \alpha_t \sum_i \delta(x_i,x_{i\rightarrow \boldsymbol{u}_i}) \\ &+ \alpha_m \sum_{i\sim j} E_m(\boldsymbol{u}_i,\boldsymbol{u}_j) \\ &+ \alpha_m \sum_{i\wr j} E_m(\boldsymbol{u}_i,\boldsymbol{u}_j) \end{aligned} \tag{5.16}$$

où x_i est une variable binaire pour le pixel i prenant des étiquettes dans $\{0,1\}$ (*fond, forme*) et $\boldsymbol{u}_i$ est un champ de vecteurs de déplacement dense comprenant des composantes horizontales et verticales $\boldsymbol{u}_i = [u_{i,x}\ u_{i,y}]^T$. Les deux composantes prennent des valeurs dans $[-T,T]$, T étant petit (3-5 pixels). Les voisinages spatial et temporel sont dénotés par, respectivement, par $\cdot\sim\cdot$ et $\cdot\wr\cdot$. Les différents α sont des poids et δ est le delta de Kronecker (voir équation (3.17)). La notation $i\rightarrow\boldsymbol{u}_i$ indique l'indice du site en suivant le vecteur de mouvement $\boldsymbol{u}_i$ à partir du pixel i, dans la frame *suivante* la frame comprenant le pixel i.

Le terme d'attache aux données E_d correspond à un seuillage assoupli, ç.à.d. il est équivalent à un seuillage avec le paramètre D si l'information *a priori* est uniforme :

$$E_d(y_i,x_i) = \begin{cases} \Delta(y_i) & \text{si } x_i = 0 \\ 2D - \Delta(y_i) & \text{si } x_i = 1 \end{cases} \tag{5.17}$$

Les termes impliquant δ régularisent le champ d'étiquettes binaires, ce qui améliore le résultat de la segmentation fond/forme. L'expression $E_m(\boldsymbol{u}_i,\boldsymbol{u}_j)$ régularise le champ de déplacement. Il s'agit d'un potentiel punissant des vecteurs de déplacement mal alignés.

Une tentative d'esquisse du graphe de dépendances de l'équation (5.16) est donnée dans la figure 5.8, à consulter de préférence en couleur. Un seul sommet de type X_i est donné (en rouge). Il est connecté à tous les sommets de son voisinage autour de lui (dans le frame suivant), puisque la régularisation se fera avec un de ces pixels, dépendant de la variable vectorielle $\mathbf{U}_i$. Chaque X_i est donc aussi connecté à chaque $\boldsymbol{U}_i$ du même indice i. Les sommets du champ $\boldsymbol{U}$ d'un voisinage spatio-temporel sont connectés entre eux. Chaque X_i et connecté à une seule observation Y_i ayant le même indice i. Chaque $\boldsymbol{U}_i$ est connecté à toutes les observations Y_j d'un voisinage spatio-temporel.

La fonction d'énergie (5.16) est difficile à minimiser à cause des interactions complexes entre x et $\boldsymbol{u}$. A lieu de la minimiser de manière approximative, e.g. par troncation d'énergie et l'algorithme α-expansion [KZ04], où par QPBO [KR07], nous avons préféré d'approximer le modèle afin de pouvoir le résoudre de manière exacte.

Nous passons à un calcul en deux étapes. Pendant le calcul de segmentation, ç.à.d. l'estimation du champ X, nous considérons les vecteurs de mouvement $\boldsymbol{u}$ comme étant constant. Ces derniers sont calculés avec un algorithme de type flot-optique, s'appuyant sur la fonction d'énergie (5.16). Dans l'optimisation du champ X, les termes impliquant E_m sont donc constant et peuvent être omis. Si $\alpha_s \leq 0$ et $\alpha_t \leq 0$, ce qui correspond à la configuration souhaitée favorisant des régions

homogènes, alors les autres termes deviendront sous-modulaires :

$$\begin{aligned}\hat{x} = \arg\min_{x} \quad & \alpha_d \sum_i E_d(y_i, x_i) \\ + \quad & \alpha_s \sum_{i \sim j} \delta(x_i, x_j) + \\ + \quad & \alpha_t \sum_i \delta(x_i, x_{i \to \boldsymbol{u}_i})\end{aligned} \tag{5.18}$$

La minimisation peut donc être calculé de manière exacte avec l'algorithme de Kolmogorov et al. [KZ04] — voir aussi le chapitre 3, section 3.4.4. Les vecteurs de mouvement $\boldsymbol{u}_i$ étant constants, ils interviennent uniquement dans le placement des arêtes dans le graphe st.

Le calcul des vecteurs de mouvement $\boldsymbol{u}_i$ à partir d'un volume d'images y_i correspond à un problème de flot optique, les méthodes existantes peuvent donc être appliquées, par exemple la méthode Sand et Teller [ST08]. Le calcul étant assez coûteux, nous avons décidé de faire une approximation en tenant compte d'une partie des termes de l'énergie (5.16). Une autre approximation sépare les composantes horizontales et verticales du champ de déplacement. En conséquence, plus aucune arête ne se trouve dans le graphe de dépendances entre les variables $u_{i,x}$ d'un coté et les variables $u_{i,y}$ de l'autre coté. L'optimisation peut donc être calculée de manière indépendante pour les deux composantes. Cela est possible en choisissant la distance L_1 (la distance de Manhattan) pour le potentiel E_m régularisant le champ $\boldsymbol{u}$ et en remplaçant la composante inconnue respective par le résultat d'une recherche locale :

$$\begin{aligned}\hat{\boldsymbol{u}} = \arg\min_{\boldsymbol{u}} \quad & \alpha_t \sum_i \min_{a \in [1,T]} |y_i - y_{i \to [\begin{smallmatrix} u_{i,x} \\ a \end{smallmatrix}]}| \\ + \quad & \alpha_t \sum_i \min_{a \in [1,T]} |y_i - y_{i \to [\begin{smallmatrix} a \\ u_{i,y} \end{smallmatrix}]}| \\ + \quad & \alpha_0 \sum_i |u_{i,x}| + \alpha_0 \sum_i |u_{i,y}| \\ + \quad & \alpha_m \sum_{i \sim j} |u_{i,x} - u_{j,x}| + |u_{i,y} - u_{j,y}|\end{aligned} \tag{5.19}$$

Les premiers deux termes favorisent une différence minimale de niveaux de gris dans la direction du mouvement. Le terme correspondant au poids α_0, qui est d'ailleurs très petit, favorise très légèrement un mouvement nul, nécessaire pour combler les effets secondaires de l'approximation séparant les composantes horizontales et verticales. Les termes de deuxième ordre favorisent donc un mouvement homogène.

Les termes de régularisation sont non sous-modulaires et ils impliquent des variables à plusieurs étiquettes, ce qui rend la méthode Kolmogorov et al. inapplicable. Par contre, l'ensemble d'étiquettes admet un ordre strict, car il s'agit de la discrétisation d'une grandeur continue, et les potentiels d'énergie sont convexes en différences d'étiquettes. L'énergie peut donc être minimisée par la méthode d'Ishikawa [Ish03] — voir aussi le chapitre 3, section 3.4.4.

5.8.2 Résultats

La méthode a été évaluée sur une base de vidéos acquise par nos partenaires dans le cadre du projet ANR Canada et contenant des scènes difficiles avec plusieurs personnes en mouvement. La figure

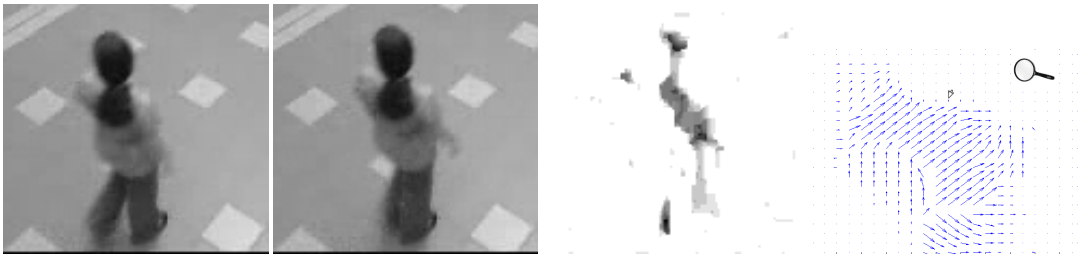


Figure 5.9 – Flot optique approximatif, de gauche à droite : frame 1, frame 2, amplitude de mouvement, vecteurs de mouvement d'une partie agrandie.

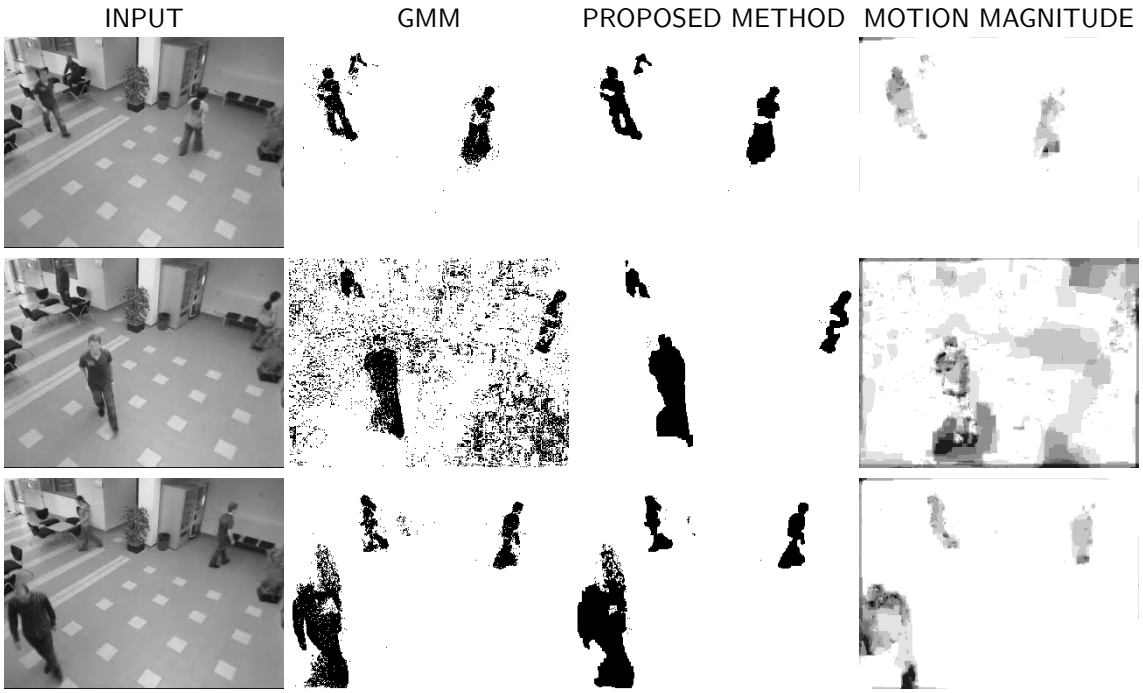
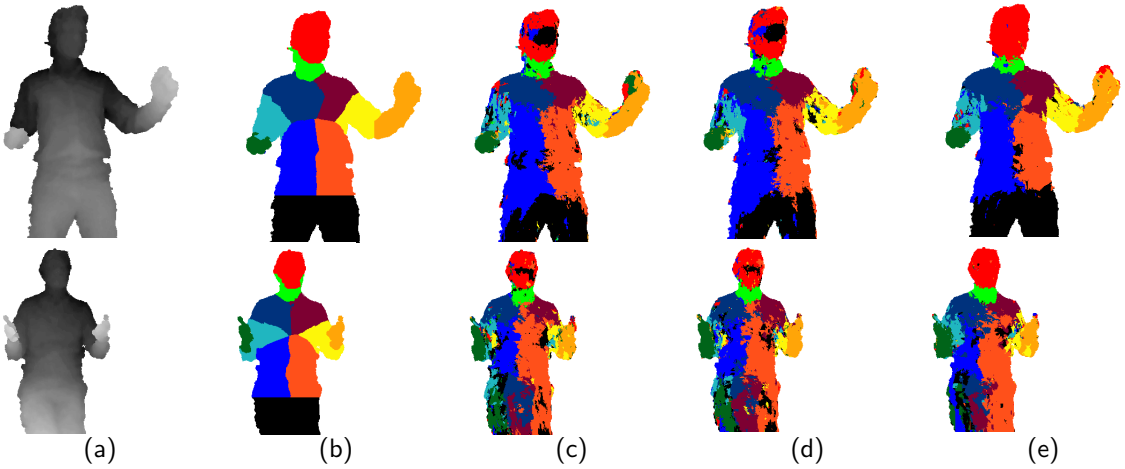


Figure 5.10 – Résultat de la soustraction de fond.

5.9 montre une zone particulière d'une paire de frames consécutives avec l'amplitude du champ de déplacement.

La figure 5.10 montre quelques résultats de la segmentation fond/forme sur nos données. Comme on peut le constater, la méthode proposée réussit à produire des résultats bien plus nets et plus précis. Plus important, les défauts dus à un changement d'illumination ont pu être corrigés. Cela n'aurait pas été possible avec des méthodes de post-traitement comme la morphologie mathématique. Un vidéo avec plus de résultats et disponible en-ligne [7].

7. http ://liris.cnrs.fr/christian.wolf/vids/bgsuboflow.gif



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

Figure 5.11 – Exemples de classification pixel par pixel : (a) image d'entrée en profondeur ; (b) vérité terrain ; (c) apprentissage classique ; (d) apprentissage spatial ; (e) apprentissage spatial (profondeur + contours).

5.9 Régularisation spatiale sans termes par paires

Les modèles présentés dans ce chapitre partagent un point commun, à savoir la segmentation d'un ensemble de pixels, d'une image ou d'une vidéo, en régions homogènes par optimisation discrète, c.à.d. par minimisation d'une fonction d'énergie comprenant des termes de régularisation. La régularisation peut améliorer les performances de manière significative, au détriment d'une complexité de calcul beaucoup plus élevée. Dans cette section nous discuterons brièvement des applications où la vitesse de calcul est un facteur primordial, par exemple l'estimation de la pose humaine pour les interfaces homme-machine. Une famille d'approches passe par une étape intermédiaire de segmentation du corps humain en parties définies préalablement, comme la main, l'avant-bras, le bras, le torse gauche, le torse droit, la tête, etc. Cela est illustré par la figure 5.11 : (a) montre deux images de profondeur en entrée et (b) montre les décompositions des images en parties obtenues manuellement par un humain.

Les approches existantes obtiennent une segmentation par une classification indépendante des pixels par des classifieurs. Le système d'estimation de pose utilisé dans le produit *Kinect* commercialisé par Microsoft passe par une classification de chaque pixel par un classifieur de type forêt aléatoire [SFC+11], ce qui permet une classification extrêmement rapide. Cette approche simple nécessite toutefois une quantité de données extraordinaire, 2.000.000.000 vecteurs pour le cas de Kinect.

Nous avons très récemment proposé une nouvelle approche d'apprentissage de classifieur adaptée à ce contexte en injectant des informations supplémentaires [JWB13]. Notre approche est applicable dans toute situation où une classification est faite sur un ensemble d'étiquettes admettant des relations spatiales entre elles. En d'autres termes, les relations spatiales que nous modélisons concernent les étiquettes directement. Dans l'exemple d'estimation de pose cité ci-dessus, l'étiquette « avant-bras » est voisin de « main » et de « bras », mais il n'est pas voisin de » tête « ou de « pied ». Cette situation n'est pas restreinte à l'estimation de la pose humaine. D'autres applications possibles sont la reconnaissance d'objets par modèles par parties [FH05, FGMR10],

	Caractéristiques	— Taux de classification — par pixel	par partie
Apprentissage classique	Profondeur	60.30	23.98
Apprentissage spatial	Profondeur	61.05	26.99
Apprentissage spatial	Profondeur + contours	**67.66**	**28.91**

Table 5.3 – Taux de classification pour des configurations différentes.

la segmentation d'objets par layout CRF [WS06] ou encore l'étiquetage sémantique de scènes naturelles [FCNL12].

Notre approche améliore l'algorithme classique d'apprentissage de forêtes aléatoires [SFC+11, LLF05], qui procède de manière itérative en apprenant le modèle de prédiction couche par couche. Pour chaque couche, l'algorithme classique choisit les paramètres et les caractéristiques en maximisant le gain en entropie mesuré sur les distributions Q des étiquettes de la couche suivante. Tout en gardant une approche pixel par pixel, notre méthode remplace l'entropie classique

$$H(Q) = \sum_k -p(k) \log p(k) \tag{5.20}$$

par l'entropie sur un nouvel alphabet Q' consistant de toutes les paires possibles (i, j) d'étiquettes de l'alphabet original, comme suit :

$$H(Q') = \sum_{i,j} -p(i)p(j) \log[p(i)p(j)] \tag{5.21}$$

En séparant les termes de (5.21) en deux groupes, un groupe pour les étiquettes voisins et un groupe pour les étiquettes non-voisins, un algorithme adaptatif peut être conçu. Nous renvoyons le lecteur à [JWB13] pour plus de détails.

Il est important de noter, que ce changement concerne uniquement l'algorithme d'apprentissage. L'algorithme de classification reste inchangé, il procède toujours de manière indépendante pour chaque pixel. Plus important, sa complexité de calcul reste inchangée. La figure 5.11 et la table 5.3 montrent les performances de l'algorithme comparé à l'apprentissage classique, mesurées par deux mesures :

- le taux de classification par pixel ;
- après une étape de regroupement des pixels en régions, c.à.d. en parties, nous appliquons une mesure introduite dans [FMJZ08] pour cette application : une région de la vérité de terrain est appariée avec une région détectée si le centre de la dernière se trouve à l'intérieur d'un cercle centré sur la région de la vérité terrain et de rayon de 50% de sa taille.

Nous pouvons constater que la performance de classification peut être améliorée de manière significative par ce nouvel algorithme d'apprentissage. Une autre contribution de ce travail est la conception de nouvelles caractéristiques basées sur les contours de l'image RGB et qui complètent les caractéristiques extraites sur l'image en profondeur. L'introduction des ces caractéristiques peut de nouveau améliorer les performances du système, comme cela est montré dans la figure 5.11e and dans le tableau 5.3.

5.10 Conclusion sur la segmentation

Dans ce chapitre nous avons présenté plusieurs modèles graphiques pour la segmentation d'images et de vidéos. Nous avons vu que ce formalisme permet d'intégrer de nombreuses informations de types différents dans le modèle, ce qui permet d'augmenter la qualité du résultat de manière significative. Les difficultés principales se situent autour des interactions plus complexes : par exemple la prise en compte de discontinuités dans le processus aléatoire, comme dans le cas des contours. De façon optimale, la régularisation devrait être désactivée pour les pixels séparés par une frontière. Traditionnellement cela passe par la désactivation de la régularisation en cas d'un gradient fort entre les sites (approche *variation totale*), ou par une modélisation directe des frontières. Cette dernière approche a l'avantage de permettre une régularisation des frontières également — voir aussi la segmentation de maillages 3D, chapitre 6.

Un autre exemple difficile d'interactions concerne l'intégration du mouvement, comme présenté dans la section 5.8 de ce chapitre. La mise en place du modèle a nécessité la séparation de la phase d'optimisation en deux étapes, une pour l'estimation des étiquettes associées aux pixels, et une pour l'estimation des étiquettes associées aux vecteurs de mouvement.

Il est vrai que les méthodes dites *graph cuts* présentent une avancée énorme dans le domaine. Un grand nombre de problèmes, considérés comme étant difficiles auparavant, peuvent être résolu de manière exacte maintenant. Toutefois, cette famille de méthodes n'est pas une "balle en argent"[8]. Il s'avère que les modèles apportant le plus d'informations à la résolution d'un problème sont en même temps les modèles dont l'optimisation est la plus difficile. Ils ne sont pas seulement NP-difficile, il est aussi souvent très difficile de trouver des bonnes solutions avec des heuristiques générales. Pourtant, la nature nous montre que la solution est possible : le système visuel humain est capable de résoudre des problèmes complexes rapidement, donc également de segmenter et de reconnaître des objets complexes.

Pour finir, nous oserons quelques prédictions des recherches à venir dans ce domaine : nous sommes convaincus que la recherche apportera quelques solutions aux problèmes suivants. Il s'agit de pistes que nous comptons également poursuivre, bien entendu :

— Nous pensons que les recherches sur les méthodes générales et applicables à tout problème seront complétées par des recherches sur des problèmes très spécifiques en vision par ordinateur. Une stratégie intéressante pourrait se baser sur des recherches locales très rapides et surtout en multi-échelle, suivies par une propagation et intégration efficace des données récoltées localement.

— Nous pensons également que la recherche apportera plus des solutions au problème de segmentation qui se passeront d'une modélisation par contraintes de deuxième ordre, comme par exemple l'ont montré les travaux sur l'estimation de pose dans le cadre du projet *Kinect* [SFC+11], et notre amélioration décrite dans la section 5.9. Nous pensons qu'il reste encore beaucoup de potentiel dans les méthodes de segmentation avec une régularisation sans algorithme d'optimisation discrète.

8. De l'Anglais *silver bullet*, une arme résolvant tous les problèmes du contexte en question

Chapitre 6

Analyse de modèles géométriques

Dans ce chapitre nous nous intéressons aux problèmes de traitement et d'analyse de modèles géométriques, sous forme de maillages surfaciques, en vue de leur optimisation (problème de remaillage) ou de leur segmentation. Quelques parallèles peuvent être établis entre la segmentation d'images et de vidéos, traitée dans le chapitre 5, et la segmentation de maillages. Les deux problèmes demandent un étiquetage d'un grand nombre de variables (pixels ou sommets) et la recherche d'un partitionnement en un ensemble de régions cohérentes. Le cas des modèles géométriques se distingue toutefois par un échantillonnage irrégulier et par la difficulté d'intégrer à la fois la géométrie et la connectivité des données.

La segmentation d'un maillage peut être motivée par plusieurs contextes applicatifs :

— l'ingénierie inverse pour produire un modèle CAO à partir du maillage d'un modèle 3D numérisé (pièces mécaniques, bâtiments, villes etc.),
— l'adaptation à la géométrie locale d'algorithmes de traitement ou de compression ;
— une étape préliminaire pour le remaillage, pour le plaquage de textures et pour l'animation ;
— l'indexation et la recherche par le contenu.

Le remaillage vise à améliorer la structure du maillage, c.à.d. la qualité des triangles et de la connectivité, tout en gardant la meilleure approximation possible de la surface. A titre d'exemples nous pouvons citer les applications suivantes :

— les simulations numériques ;
— la compression ;
— le rendu ;
— l'amélioration de la qualité de modèles bruités (lissage et débruitage).

6.1 Contexte, projets et collaborations

Les travaux décrits ici sont nés d'un souhait d'étendre nos travaux sur les images aux modèles géométriques, et donc sur une grille irrégulière. Ensemble avec Guillaume Lavoué du LIRIS [LW08], les premiers travaux ont été réalisés sur la segmentation de maillages en utilisation l'information de courbure, brièvement décrits dans la section 6.2.1. Les résultats ont rapidement confirmé l'intérêt d'une modélisation globale pour l'analyse de maillages. Une thèse de type « allocation de recherche fléchée sur le sujet » a été déposée, et acceptée, avec Florent Dupont du LIRIS, conduisant au recrutement de Vincent Vidal en octobre 2008. Ayant soutenue sa thèse en décembre 2011, V.

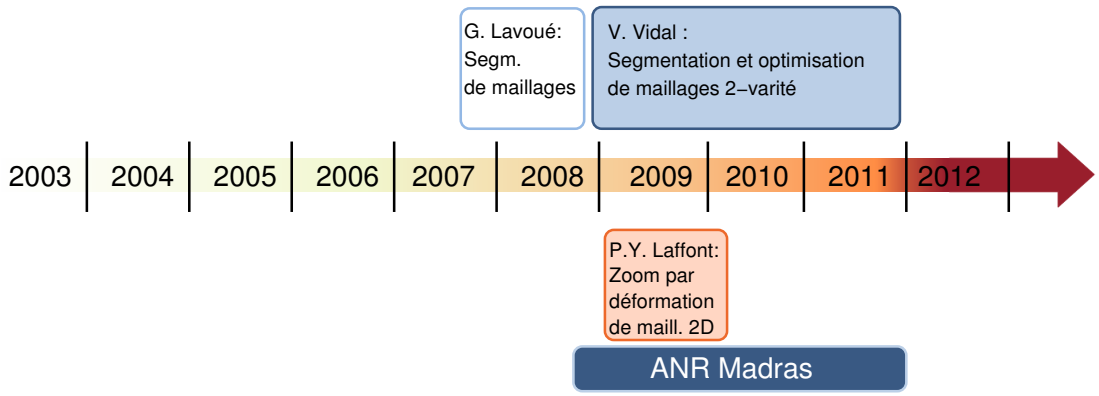


Figure 6.1 – Illustration de l'évolution temporelle des travaux du chapitre 6.

Vidal a été recruté comme Maître de Conférences dans l'équipe M2Disco du LIRIS pour une prise de ses fonctions en septembre 2012. La thèse a traité les problèmes de segmentation de maillages et de remaillage, décrits dans les deux sections 6.2 et 6.3, respectivement.

Ces travaux font également partie du projet ANR MADRAS (2008-2010) porté par Florent Dupont du LIRIS avec les partenaires LIFL et INRIA Rhône-Alpes [1]. Son objectif était de créer une collection de maillages 3D et 3D+t et de mettre en œuvre des nouveaux algorithmes de segmentation.

Dans le cadre d'une collaboration avec Sung-Eui Yoon du KAIST en Corée, et du stage M2R de Pierre-Yves Laffont, nous avons travaillé sur le *retargeting* d'images, c.à.d. le recadrage d'images avec prise en compte du contenu. Le problème a quelques similarités avec le remaillage surfacique 3D, s'agissant d'optimiser la déformation d'un maillage en minimisant des mesures de bruit induit. Ici le bruit est toutefois lié aux dégradations causées dans une image 2D. Ces travaux seront brièvement discutés dans la section 6.4.

La figure 6.1 montre les différents travaux sur un axe temporel ensemble avec les projets dans lesquels ils ont été effectués.

6.2 Segmentation et décomposition de maillages

La segmentation et la décomposition de maillages traitent le problème de partitionner un maillage en régions selon un certain critère d'homogénéité. Les travaux existants sont généralement classés en deux groupes [AKM+06, Sha08] :

- les travaux basés sur la géométrie de bas niveau, regroupant des régions par normales, par courbures et/ou par d'autres mesures similaires. Souvent l'objectif est de produire une nouvelle représentation donnant une bonne approximation de la surface. Un exemple typique est l'ingénierie inverse, où le maillage est décomposé en primitives géométriques simples, l'objectif étant de trouver les primitives étant à l'origine de la conception du modèle 3D ;
- les travaux cherchant une décomposition en régions de plus haut niveau, proche de la « sémantique ». Le critère de homogénéité dépend de l'application en question, par exemple l'indexation de modèles 3D par parties [TVD07, GCO06, SSS+09, TCF10]. Dans certaines

1. http://www-rech.telecom-lille1.eu/madras

applications spécifiques, par exemple la visualisation scientifique, la méthode peut recourir à des attributs liés à l'application, comme des températures ou des densités.

Les mesures géométriques les plus utilisées sont basées sur les attributs tels que la courbure [CSM03], la rugosité [Gui07], la saillance [LVJ05], les caractéristiques topographiques [LZH+07], la *Shape Diameter Function* (une mesure liée au concept de l'axe-médian) [SSSCO08] etc.

Certaines méthodes sont basées sur le clustering des faces pour obtenir des régions homogènes, soit planaires [CSAD04], soit des régions de courbure constante [LPRM02, LDB05]. D'autres méthodes recourent aux points d'intérêt [KLT05], au squelette [TVD06], et à l'analyse spectrale [LZ04].

Quelques méthodes rares sont basées sur une optimisation globale. [KT03] segmente un maillage en 2 régions distinctes plus une régions de frontière (région « je ne sais pas »). Le clustering est calculé à l'aide d'un algorithme de type flou avec un post traitement pour raffiner les bords basé sur les *graph cuts*. Dans [SN09], un CRF est utilisé pour régulariser la segmentation, similaire à notre méthode présentée dans la section 6.2.2. La différence avec notre méthode réside dans les termes d'attache aux données. Les termes unaires dans notre approche mesurent l'erreur d'approximation, important dans notre contexte applicatif de rétro-ingénierie, tandis que les termes unaires dans [SN09] intègrent des connaissances de type sémantique apprises à partir de plusieurs segmentations d'exemple. D'autres méthodes basées sur l'apprentissage à partir d'un corpus de modèles segmentés sont [KH10] et [BLVD11], la dernières ayant la particularité d'apprendre les frontières entre les régions.

Pour les méthodes ciblant l'ingénierie inverse, la segmentation ciblée est une décomposition telle que chaque région puisse être approximée par une primitive géométrique parmi un ensemble de types possibles (plans, sphères, cylindres, cônes etc.). Comme souvent, ce problème est mal posé puisque deux étapes sont mutuellement inter-dépendantes :

— le choix d'un type de primitive et le *fitting* de ses paramètres pour une région donnée dépendent de la segmentation du maillage en régions ;
— la segmentation du maillage nécessite une affectation des sommets aux primitives qui dépend des paramètres de ces dernières.

La plupart des approches existantes traitent ce problème de manière locale ou de manière gloutonne. Les approches variationnelles, dont l'exemple incontournable est VSA (*Variational Shape Approximation*) [CSAD04], décomposent le maillage en minimisant une mesure d'erreur géométrique entre un triangle et sa primitive (appelée « *proxy* »). L'algorithme procède par i) la sélection d'un ensemble de points de démarrage (*seed points*) ; ii) une étape de croissance de régions autour des points de démarrage minimisant une erreur d'approximation. A l'origine conçu pour des proxies planaires, ce système a été étendu aux proxies de type sphère et cylindre [WK05] et aux proxies de type surfaces quadratiques [YLW06]. Plusieurs méthodes sont basée sur la variation totale [DPFH10, RBB+12, ZZWC12], un concept connu de la segmentation d'images — voir une description plus détaillée sur la page 148.

Le *Hierarchical Face Clustering* [AFS06] partage avec les approches variationnelles le principe d'optimisation gloutonne. Démarrant avec une décomposition initiale où chaque triangle correspond à un proxy, l'algorithme fusionne des proxies en minimisant l'erreur d'approximation gérée par une file de priorité, les priorités étant inversement proportionnelles aux erreurs. De manière similaire, dans [GWH01], une hiérarchie de proxies est créée à partir de contractions d'arêtes.

Toute une famille de méthodes de décomposition de nuages de points est basée sur l'algorithme RANSAC (*Random Sampling and Consensus*), dont l'objectif est d'estimer les paramètres d'un

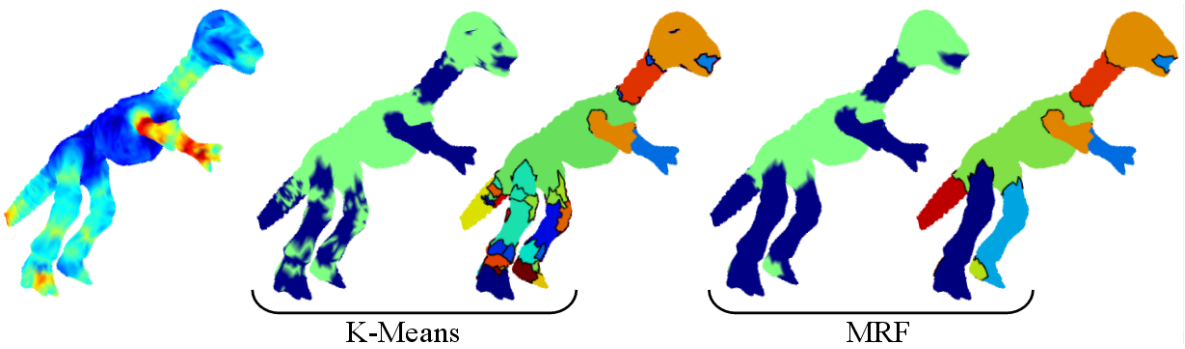


Figure 6.2 – Segmentation par courbure, de gauche à droite : le champ de courbure ; clustering par k-moyennes ($k = 2$) ; résultat de la croissance de régions ; segmentation des sommets par MRF ; résultat de la croissance de régions.

modèle paramétrique de manière optimale avec prise en compte des valeurs aberrantes [FB81]. L'algorithme itératif alterne des phases d'échantillonnage de points de support avec une estimation de paramètres (souvent par des critères de moindres carrés) et une phase de sélection de points aberrants. Le modèle sélectionné est celui ayant le nombre de points de support le plus élevé. Des versions efficaces basées sur des structures de données accélératrices ont été proposées plus récemment, parmi lesquelles celle de Schnabel et al. est sans doute la plus connue [SWK07].

Un problème similaire dans le domaine de l'analyse d'images est traité dans [TBKL12]. Dans un contexte d'images de formes produits par des humains, typiquement des scènes urbaines, une image est décomposée en segments de lignes satisfaisant des contraintes globales, telle que la présence de points de fuites, le placement des points de fuites des lignes horizontales sur une seule ligne etc. Le problème est discrétisé et résolu par des méthodes d'optimisation discrète.

Plus compléter nous mentionnons aussi deux cas d'usage un peu particuliers :

— les méthodes non automatiques, dites supervisées ou semi-supervisées, où la segmentation est guidée ou au moins amorcée par un utilisateur humain, par exemple [ZT10, BAT12] ;
— les méthodes segmentant plusieurs objets à la fois en gardant une certaine cohérence entre les résultats, par exemple [SvKK$^+$11, HKG11].

Dans les deux sous-sections suivantes nous présenterons brièvement nos travaux sur les deux familles décrites ci-dessus, à savoir la segmentation de maillages par courbure (voir la sous-section 6.2.1) et la décomposition de maillages en primitives géométriques (voir la sous-section 6.2.2).

6.2.1 Segmentation de maillages

Dans [LW08] nous avons proposé une nouvelle méthode de segmentation de maillages basée sur les MRF. L'objectif était d'améliorer les méthodes classiques procédant par un clustering des sommets [LDB05] en ajoutant une régularisation spatiale. Les méthodes classiques procèdent par un clustering des sommets du maillage dans l'espace des caractéristiques (courbure, rugosité etc.) suivi par une étape de croissance de régions pour obtenir les régions composées de faces, comme demandé par la plupart des applications dans ce contexte. Une régularisation spatiale permet de diminuer les effets du bruit sur la segmentation.

Notre méthode modélise le maillage par un MRF défini sur les sommets du maillage. Le graphe

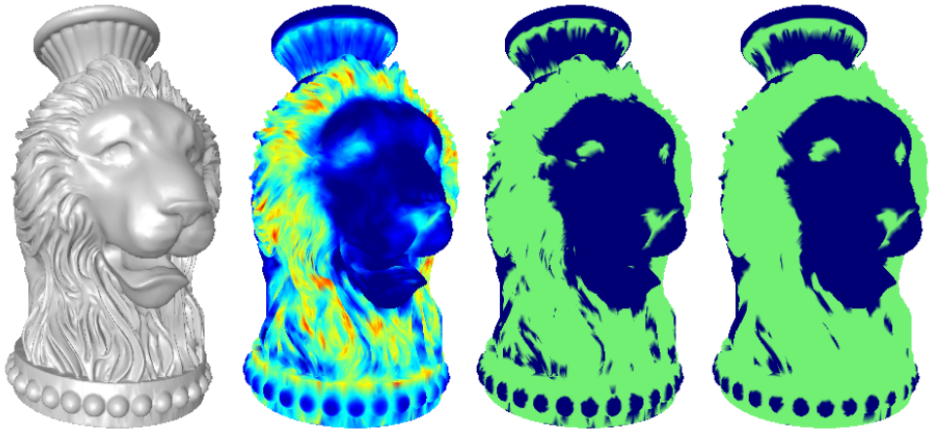


Figure 6.3 – Segmentation par rugosité, de gauche à droite : le champ de rugosité ; clustering par k-moynnes ($k = 2$) ; segmentation des sommets par MRF.

de dépendances du MRF correspond à la connectivité du maillage. La fonction d'énergie est définie sur les variables cachées v_i correspondant aux étiquettes et sur les variables observées g_i correspondant aux caractéristiques géométriques locales. Elle est composée de termes d'attache aux données $E_1(.,.)$ et de termes pour le prior $E_3(.,.,.)$:

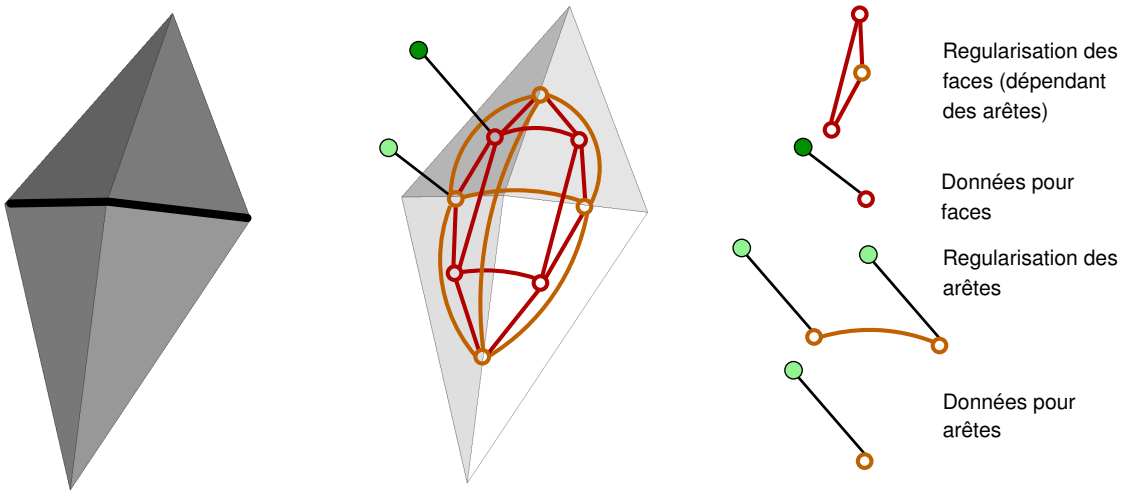
$$E(v;g) = \sum_{i \in \mathcal{S}^{(s)}} E_1(v_i, g_i) \; + \; \beta \sum_{(i,j,k) \in \mathcal{S}^{(t)}} E_3(v_i, v_j, v_k) \tag{6.1}$$

Ici $\mathcal{S}^{(s)}$ est l'ensemble des sommets, $\mathcal{S}^{(t)}$ est l'ensemble des triangles et β est un poids. Les termes d'attache aux données sont modélisés comme une variation Gaussienne d'une grandeur dépendant de l'application (courbure, rugosité etc.). Les termes de régularisation sont basés sur le modèle de Potts, voir aussi le chapitre 3, section 3.2.2, et d'autres applications de ce modèle sur la segmentation d'images présentées dans le chapitre 5. La fonction d'énergie (6.1) peut être minimisée avec les méthodes classiques, comme le recuit simulé utilisé dans nos expériences. Le modèle étant probabiliste de type génératif, l'estimation des hyper-paramètres peut se faire par moindres carrés comme décrit dans le chapitre 5, section 5.6.

Le gain obtenu par la régularisation spatiale est significatif, comme on peut le constater dans les figures 6.2 et 6.3. La segmentation n'est pas perturbée par des variations locales de la caractéristique choisie pour la segmentation, les petites régions aléatoires ont été éliminées.

6.2.2 Décomposition de maillages

Nous avons proposé une solution au problème de l'ingénierie inverse, qui consiste à chercher une décomposition globalement cohérente du maillage en intégrant plusieurs contraintes [VWD12b]. Une propriété particulièrement intéressante de notre méthode est l'intégration de la décomposition du maillage avec une extraction des arêtes vives, c.à.d. des lignes correspondant aux discontinuités de courbure, et qui correspondent donc avec une grande probabilité aux frontières entre les parties de l'objet. La figure 6.4a illustre ce principe en utilisant l'exemple d'un morceau de maillage comprenant quatre triangles et une arête vive traversant le maillage horizontalement.



(a) (b) (c)

Figure 6.4 – Décomposition de maillages : (a) un extrait d'un maillage avec une arête vive ; (b) le graphe de dépendances correspondant. Pour des raisons de simplicité, uniquement deux observations sont montrées ; (c) Les types de cliques utilisés dans la fonction d'énergie.

L'intégration d'un processus dédié pour les frontières entre les régions a été introduite relativement tôt dans le domaine de la restauration d'images [GG84]. L'interaction entre les deux processus respectifs, pixels et frontières, a comme objectif de préserver les contours dans l'image. La complexité de l'interaction a heurté la popularité du formalisme, la communauté lui préfère souvent celui de la variation totale (TV), également initialement développée pour la restauration d'images [ROF92], et que nous expliquerons brièvement ci-dessous. Dans ce modèle, l'image cachée u est obtenue par une minimisation de l'intégrale de ses variations :

$$\min_u \left\{ \int_\Omega |\nabla u| d\Omega \right\} \tag{6.2}$$

où Ω est la grille de l'image. Le minimum global étant obtenue par une solution triviale (une image constante), cette minimisation est effectuée sous une contrainte liant la solution à l'image d'entrée f :

$$\int_\Omega (u - f)^2 d\Omega = \sigma^2 \tag{6.3}$$

où σ^2 est la variance du bruit ayant dégradé l'image, supposé connu. L'adaptation à la segmentation d'images donne le modèle Mumford-Shah [MS85] :

$$\min_u \int_\Omega |u - f|^2 d\Omega = \sigma^2 + \alpha \int_{\Omega \setminus \Gamma} |\nabla u|^2 d\Omega + \nu \text{ Length}(\Gamma) \tag{6.4}$$

Ici, l'image de résultat u est considérée lisse par morceau, les morceaux étant séparés par les frontières Γ. Le terme d'attache aux données mesure la différence avec l'image d'entrée f, le deuxième terme mesure l'homogénéité des régions obtenues et le troisième terme minimise la longueur des frontières. La minimisation de ces énergies passe pour la plupart du temps par la

solution d'une équation de type Euler-Lagrange résolue par descente de gradient ; des solutions par *graph cuts* ont été proposées [DS06].

Récemment, plusieurs articles ont traité la segmentation de maillages surfaciques 3D à l'aide du principe de la variation totale. Dans [DPFH10], un maillage est segmenté par rapport à un champ de scalaires ou un champ de vecteurs. Le terme de variation totale est défini sur une paramétrisation locale du maillage, l'énergie est minimisée par gradient projeté. Un maillage est segmenté par rapport à des attributs spectrales issues de la matrice Laplacienne dans [ZZWC12]. Dans [RBB$^+$12], des maillages dynamiques sont décomposés en segments de mouvement rigide.

Pour les travaux décrits dans ce mémoire nous avons préféré le framework d'un modèle graphique avec un processus dédié sur les frontières. Plusieurs raisons nous ont emmené à ce choix. Tout d'abord, un processus dédié sur les frontières permet la régularisation de ces dernières et donc une plus grande précision. Deuxièmement, et contrairement aux approches variationnelles, une solution par *graph cuts* nous permet de sélectionner le nombre de régions de manière automatique. La segmentation de maillages par variation totale est néanmoins une alternative dont nous envisageons l'étude dans l'avenir.

Modélisation

Nous souhaitons intégrer les informations suivantes dans le processus de segmentation :

- A chaque triangle sera affecté un proxy, dont le type sera parmi les suivants : plan, sphère, cylindre. Un des critères pour le choix du type et des paramètres est l'erreur d'approximation de la surface ;
- Pour chaque arête nous déterminerons si cette arête est vive ou pas. Un des critères est la géométrie locale — entre autres, une estimation de l'angle dièdre entre les deux plans séparés par l'arête, et d'autres mesures ;
- Le modèle favorisera l'extraction d'arêtes vives « continues » ; il évitera donc des arêtes vives isolées et il favorisera des enchainements d'arêtes vives ;
- Le modèles favorisera une égalité de proxies pour des triangles voisins séparés par des arêtes non-vives, par exemple les deux triangles en haut de l'arête vive dans la figure 6.4a ;
- De la même manière, le modèle favorisera le choix de proxies différents pour des triangles voisins séparés par une arête vive, par exemple les deux triangles gauches dans la figure 6.4a ;

Tous ces critères seront pris en compte de manière globale par un modèle graphique sur un champ de variables de différents types. Le problème original est de nature continue-discrète, les paramètres des différents proxies étant continus. Nous avons procédé par une approche classique qui consiste à discrétiser le problème : un ensemble de proxies candidats est créé, dont les paramètres sont estimés et fixés. Ensuite, l'algorithme alterne entre deux étapes différentes, illustrées dans la figure 6.5 :

- le choix d'un proxy pour chaque triangle et une décision pour chaque arête (vive ou non-vive). Cet étiquetage est un problème combinatoire que nous résolvons par minimisation d'une fonction d'énergie discrète ;
- pour un étiquetage donné, l'estimation des paramètres de chaque proxy à partir des triangles qui lui ont été associés.

Le modèle graphique de ce problème comprend donc des observations basées sur la géométrie du maillage, ainsi que des variables cachées :

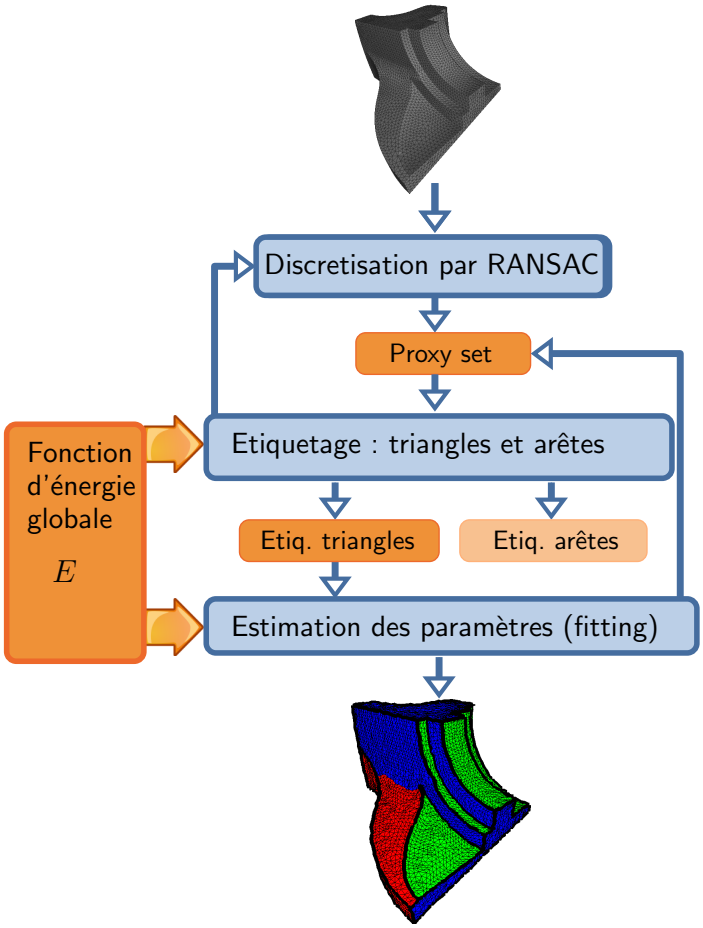


Figure 6.5 – Un schéma illustrant la méthode de décomposition de maillages.

— une variable cachée discrète x_i est associée à chaque arête i de l'ensemble $\mathcal{S}^{(x)}$ des arêtes du maillage. $x_i{=}1$ si l'arête i est vive, et $x_i{=}0$ sinon ;
— une variable cachée discrète t_i est associée à chaque triangle i de l'ensemble $\mathcal{S}^{(t)}$ de triangles. La variable peut prendre des valeurs correspondant aux différentes proxies : $\mathcal{L}^t = \{\varnothing\} \cup \{0, \ldots, m-1\}$, où m est le nombre de proxies. Un proxy $\varnothing$ est ajouté indiquant la non-association du triangle en question à un proxy de l'ensemble. En d'autres termes, le triangle est considéré comme aberrant ;
— une variable g_i associée à chaque triangle i dont les valeurs correspondent à sa géométrie ;
— une variable y_i associée à chaque arête i dont les valeurs correspondent à des mesures intervenant dans l'estimation si, oui ou non, cette arête est vive ;
— une variable $\mathcal{P}_i$ associée à chaque proxy de forme i, et dont les valeurs correspondent aux paramètres du proxy. Les indices i correspondent aux étiquettes des variables $x_.$.

La fonction d'énergie à minimiser associée à ce modèle comprend plusieurs termes :

$$\begin{aligned} E(t;x;g;y,\theta,\phi;\mathcal{P}) \quad &= \\ \kappa \quad &\sum_{i\in\mathcal{S}^{(x)}} \quad E_{feat}(x_i;y) \\ +\lambda \quad &\sum_{(i,j)\in\mathcal{E}^{(x)}} \quad E_{dir}(x_i,x_j;\theta,\phi) \\ +\rho \quad &\sum_{i\in\mathcal{S}^{(t)}} \quad E_{approx}(t_i;g_i;\mathcal{P}_{t_i}) \\ + \quad &\sum_{(i,j)\in\mathcal{E}^{(t)}} \quad E_{reg}(t_i,t_j,x_{\psi(i,j)}) \\ + \quad & \quad E_{cout}(t) \end{aligned} \tag{6.5}$$

Dans ce qui suit, nous décrivant les grandes lignes de ce modèle. Pour les détails nous renverrons le lecteur à [VWD12b] et à [VWD11]. Il est important de noter que le graphe de dépendances n'est pas équivalent à la structure du maillage, ce qui s'explique par le fait que les variables du problème ne correspondent pas aux sommets du maillage mais aux arêtes et aux triangles. La figure 6.4b montre une partie du graphe de dépendances pour le morceau de maillage montré dans la figure 6.4a. Les différents cliques utilisées sont montrées dans la figure 6.4c. Ces cliques correspondent aux différents types de termes de la fonction d'énergie (6.5).

L'erreur de la décomposition est contrôlée par les termes $E_{approx}(t_i;g_i;\mathcal{P}_{t_i})$ qui dépendent, pour chaque triangle $i \in \mathcal{S}^{(t)}$, de l'étiquette t_i choisissant le proxy, de la géométrie $\mathcal{P}_{t_i}$ du proxy et de la géométrie g_i du triangle même :

$$E_{approx}(t_i;g_i;\mathcal{P}_{t_i}) = \begin{cases} \mathcal{L}^2(g_i,\mathcal{P}_{t_i}) & \text{si } \mathcal{P}_{t_i} \neq \varnothing \\ \gamma \times |g_i| & \text{sinon} \end{cases} \tag{6.6}$$

où $\mathcal{L}^2(g_i,\mathcal{P}_{t_i})$ est une métrique d'erreur entre le proxy $\mathcal{P}_{t_i}$ et la projection du triangle g_i sur ce dernier. Si le triangle n'est pas affecté à un proxy, une punition proportionnelle à son aire est ajouté à l'énergie. Ces termes vont pousser la solution vers une bonne approximation de la surface d'origine.

Une bonne estimation des arêtes vives améliora également la décomposition du maillage en primitives grâce aux interactions entres les arêtes et les triangles. Dans ce contexte, nous considérons comme « vive » une arête à la frontière entre deux primitives géométriques, donc entre deux proxies différents. Le terme d'interaction entre arêtes et triangles intégre ce type de connaissances. Il est défini pour toutes les paires de triangles voisins i et j ainsi que l'arête $\psi(i,j)$ qui les sépare :

$$E_{reg}(t_i,t_j,x_{\psi(i,j)}) = \begin{cases} \mu & \text{si } t_i \neq t_j \text{ et } x_{\psi(i,j)} = 0 \\ \nu & \text{si } t_i \neq t_j \text{ et } x_{\psi(i,j)} = 1 \\ \zeta & \text{si } t_i = t_j \text{ et } x_{\psi(i,j)} = 0 \\ \eta & \text{si } t_i = t_j \text{ et } x_{\psi(i,j)} = 1 \end{cases} \tag{6.7}$$

où μ, ν, ζ, et η sont des constantes qui contrôlent le comportement de ces termes. En mettant $\mu = \nu \geq 0$ et $\zeta = \eta = 0$, nous retombons sur le modèle de Potts/Ising également utilisé par l'algorithme PEaRL pour la segmentation de nuages de points [IB11]. Notre objectif est de favoriser l'égalité de labels pour des triangles séparés par une arête normale, et de favoriser la différence de labels pour des triangles séparés par une arête vive. Cela nous emmène à une paramétrisation

satisfaisant $\eta \geq \mu, \nu, \zeta,\ \mu = \nu > 0$ et $\zeta = 0$. Notons que cela rend les termes E_{reg} non sous-modulaires.

La minimisation d'une fonction comprenant les deux termes E_{approx} et E_{reg} uniquement aurait peu d'avantages comparée à l'utilisation des termes E_{approx} uniquement, car la régularisation faite par les termes E_{reg} s'appuie sur une estimation d'arêtes « dans le vide » : l'information passée des arêtes vers les triangles est à peu près équivalent à l'information passée des triangles vers les arêtes. Deux autres types d'interaction sont donc ajoutés au modèle. La première interaction est définie sur des couples i et j d'arêtes voisines, elle favorise des lignes contiguës d'arêtes vives :

$$E_{dir}(x_i, x_j; \theta, \phi) = \begin{cases} 0 & \text{si } x_i \neq x_j \\ -\tau & \text{si } x_i = x_j = 0 \\ -\Gamma(\theta_i, \theta_j, \phi_{\psi'(i,j)}) & \text{sinon} \end{cases} \tag{6.8}$$

où τ est une constante positive contrôlant l'homogénéité des arêtes normales. Les arêtes voisines normales sont donc favorisées par rapport à des arêtes voisines de types différents (normale + vive). L'énergie de deux arêtes voisines de types vive dépend de la géométrie locale du maillage, en particulier des deux angles dièdres θ_i et θ_j associés aux deux arêtes respectives, et de l'angle tangentiels $\phi_{\psi'(i,j)}$ entre les deux arêtes. La fonction Γ est une fonction positive favorisant des arêtes alignées avec des angles dièdres similaires (voir [VWD11]).

L'ajout d'un terme d'attache aux données associé aux arêtes permet de guider l'estimation des arêtes vives. Bien évidemment notre définition d'une arête vive comme étant une arête sur une frontière entre deux proxies n'est pas directement liée à la géométrie locale. Une frontière entre un cylindre et un plan, par exemple, ne pourra pas être détectée facilement par la géométrie (très) locale si le plan est tangent au cylindre. Néanmoins, la majorité des cas profite de l'injection d'informations basées sur la géométrie.

Pour pouvoir tenir compte de la grande variabilité des situations dans lesquelles une arête vive peut séparer des primitives géométriques, notre terme d'attache aux données est basé sur l'apprentissage automatique par un classifieur. Pour chaque arête vive, 43 caractéristiques géométriques sont calculées, dont certaines sur plusieurs échelles. Elles sont basées sur la courbure, sur l'angle dièdre, et sur un nouvel estimateur de l'angle entre les plans tangents basé sur une segmentation locale robuste. Pour plus de détails nous renvoyons le lecteur à [VWD11].

Un modèle de prédiction de type SVM « Séparateur à Vaste Marge » est appris sur un ensemble de données d'apprentissage annoté manuellement. Pour une arête i ayant les caractéristiques y_i, le séparateur donne la prédiction sous la forme d'une réponse signée $d(y)$, dont le signe représente la décision et la valeur absolue la confiance, c.à.d. le distance du vecteur de caractéristiques y_i au hyper-plan séparant les deux classes. L'intégration dans le modèle graphique est assez directe :

$$E_{feat}(x_i; y) = \exp\left\{ (-1)^{x_i} d(y) \right\} \tag{6.9}$$

Les termes décrits ci-dessus permettent la modélisation des solutions globalement cohérentes qui expliquent bien les données observées : position et orientation des triangles, courbures, angles dièdres etc. Les solutions correspondent à des étiquetages des triangles et des arêtes. Pour beaucoup d'applications la partie intéressante est l'étiquetage des triangles, qui correspond à la décomposition du maillage en primitives. Rappelons ici la nature discrète de la méthode : une discrétisation initiale par RANSAC va créer un ensemble de proxies candidats, donc certains — pas nécessairement tous

— seront choisis par l'étiquetage optimal. Après l'étiquetage suivra une ré-estimation des paramètres des proxies, la suppression de certains proxies et l'ajout d'autres proxies (voir [VWD12b]). Un des objectifs de la décomposition est la compacité. Nous souhaitons obtenir un nombre faible de proxies utilisés, tout en gardant une faible erreur d'approximation et une régularité de la solution.

Un dernier terme, inspiré du coût de l'étiquetage proposé dans [DOIB10], a été donc ajouté au modèle. Utilisé dans la phase d'étiquetage, son rôle est de contrôler le nombre de proxies utilisés parmi tous les proxies mis à disposition au système :

$$E_{cout}(t) = \sum_{p \in \mathcal{P}} \beta_p \begin{cases} 1 & \text{si } p \text{ a été choisi parmi les triangles } t \\ 0 & \text{sinon} \end{cases} \tag{6.10}$$

La pénalité β_p peut être différente pour chaque proxy et dépend en réalité de son type seulement. Ainsi, les proxies plus simples, comme les plans, sont favorisés par rapport aux proxies plus complexes, comme les sphères. Ce terme, qui poussera la décomposition vers une solution simple utilisant peu de proxies et de types simples, sera contrebalancé par le terme de fidélité, donné dans l'équation (6.6), qui guidera la décomposition ver une solution avec un grand nombre de proxies afin de la rendre plus fidèle à la surface d'origine.

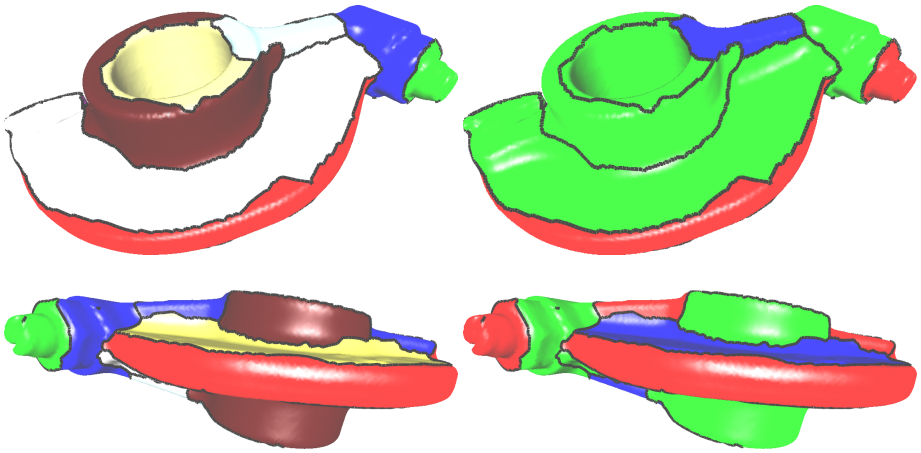
Notons que les termes de coût dans l'équation (6.10) transforment la structure du modèle graphique en un modèle non Markovien, c.à.d. le modèle probabiliste dont la fonction d'énergie est donnée par (6.5) n'est plus Markovien. Pour être plus précis, ce modèle est Markovien dans le sens trivial uniquement, puisqu'il contient des cliques contenant tous les sommets du graphe. En général, la non-Markovianité a un désavantage lors de la minimisation de la fonction d'énergie, parce que la grande majorité des algorithmes de minimisation s'appuient sur elle pour accélérer le calcul. Dans ce cas concret, les termes dans (6.10) sont graph représentables et peuvent être inclus dans un st-graph pour une minimisation par *graph cuts* [DOIB10].

Minimisation

Comme mentionné ci-dessus, les termes de cohérence E_{reg} définis dans l'équation (6.7) rendent la fonction d'énergie non sous-modulaire. Cela peut être vu intuitivement et rapidement pour le cas où une arête séparant deux triangles est vive. Le fonctionnel favorise alors l'inégalité des étiquettes des triangles, ce qui est à l'encontre de la sous-modularité, qui demande de favoriser l'homogénéité. Plus formellement, si le fonctionnel $E_{reg}(t_i, t_j, x_{\psi(i,j)})$ de trois paramètres est transformé en un fonctionnel de deux paramètres en fixant $x_{\psi(i,j)} = 1$, alors la condition de sous-modularité n'est pas satisfaite — voir l'équation (3.33) définie dans le chapitre 3, section 3.4.4.

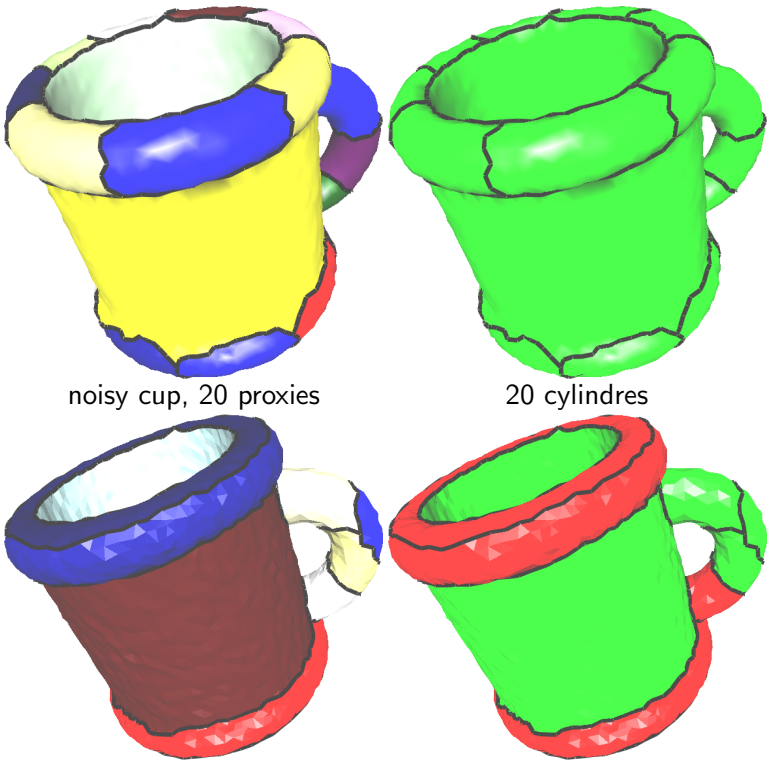
Aucune méthode polynomiale n'est connue pour la minimisation de ce type de fonctions. Les méthodes approchées les plus appropriées sont probablement la propagation de croyances (BP) et QPBO [KR07] de la famille des *graph cuts* (voir aussi le chapitre 3), les deux ayant des forces et des faiblesses différentes. La qualité de la solution obtenue par la BP dépend fortement de la structure du graphe, alors que la qualité de la solution obtenue par QPPO dépend surtout du nombre de termes non sous-modulaires de la fonction à minimiser. Une différence majeure est l'absence de toute garantie de convergence pour BP, à l'exception des cas où les graphes sont extrêmement simples, alors que QPBO garantit la convergence vers un optimum local, qui très souvent est de très bonne qualité.

Dans notre cas, le nombre de arêtes normales est largement supérieur au nombre d'arêtes vives. Comme les termes de cohérence sont sous-modulaires pour les arêtes normales, nous avons choisi



rockerArm 80K triangles 2 planes, 3 sphères, 7 cylindres

Figure 6.6 – Résultats de la décomposition de maillages : (à gauche) une couleur par proxy ; (à droite) une couleur par type de primitive {bleu = plan, rouge = sphère, vert = cylindre}



noisy cup, 20 proxies 20 cylindres

noisy cup, 10 proxies 5 sphères, 5 cylinders

Figure 6.7 – Résultats de la décomposition de maillages : (à gauche) une couleur par proxy ; (à droite) une couleur par type de primitive {bleu = plan, rouge = sphère, vert = cylindre} ; (en haut) le coût pour les sphères est supérieur au coût pour les cylindres ; (en bas) le coût pour les cylindres est supérieur au coût pour les sphères ;

(a) Notre méthode — métrique $\mathcal{L}^2$ (b) HFC — métrique $\mathcal{L}^2$ (c) VSA — métrique $\mathcal{L}^2$

Figure 6.8 – Décomposition d'une sphère en un ensemble de plans : comparaison avec VSA et HFC.

QPBO pour la minimisation, avec un mouvement d'expansion α pour la gestions des étiquettes (voir aussi le chapitre 3). Dans nos expériences nous avons remarqué que l'estimation des étiquettes des triangles profite de manière significative de l'estimation des étiquettes des arêtes ; Par contre, l'inverse ne s'est pas réalisé. La décomposition du maillage n'a guerre d'effet sur l'estimation des arêtes vives. Cela est probablement du à la non sous-modularité des termes de cohérence et à l'incapacité de QPBO de gérer cela. En conséquence, nous avons séparé l'estimation des deux champs d'étiquettes, les arêtes étant estimées avant les triangles.

Résultats

Nous avons appliqué la méthode à un grand ensemble d'objets connus dans le domaine et d'origines différentes. Certains objets sont issus de numérisation. Faute de place, nous donnerons un aperçu forcement incomplet des résultats. Deux vues du maillage « rocker arm » sont montrées dans la figure 6.6, montrant la bonne qualité de la décomposition. La table 6.1 donne quelques statistiques, surtout le nombre de primitives estimées sur quelques objets connus ainsi que quelques objets standards correspondant à des versions bruitées de nos primitives (sphère, cylindre).

Dans la majorité de nos expériences, les poids β_p ont été configurés pour favoriser des cylindres plutôt que des sphères. Un résultat sur un échange de cette préférence est donné dans la figure 6.7, où l'on peut voir deux résultats obtenus à partir de deux préférences différentes. La configuration par défaut produit une décomposition fidèle pour une grande majorité de cas. Le système est néanmoins souple, il se laisse adapter à des besoins spécifiques.

La figure 6.8 montre une sphère bruitée décomposée en plans par trois méthodes différentes : par VSA [CSAD04], par HFC [AFS06], et par notre méthode. Pour HFC et pour notre méthode, les proxies ont été restreints aux plans, VSA est naturellement restreint aux plans. L'objectif de cette expérience atypique a été de montrer la supériorité de notre méthode par rapport à des méthodes non globales dans des situations complexes où la surface s'écarte des proxies traités. Comme on peut le voir, VSA et HFC se laissent guider dans la décomposition par le bruit, ce qui produit des frontières non lisses et très aléatoires, tandis que les frontières produites par notre méthode sont assez lisses.

Modèles	#t (K)	#e (K)	Temps (sec)	#p	#s	#c
cylinder	2.36	3.54	7	2	0	1
screw	2.48	3.72	5	4	0	3
joint	6.02	9.02	38	9	0	3
sphere	9.80	14.70	40	0	1	0
cup	11.34	17.01	89	0	0	20
fandisk	12.99	19.48	102	8	2	6
carter*	32.42	48.63	851	1	6	6
rockerArm*	4.78	7.18	50	2	3	7
rockerArm	80.35	120.53	3130	2	3	7
hand	11.69	17.54	207	0	1	12
bunny	69.67	104.50	2324	2	6	2

Table 6.1 – Quelques statistiques sur les ségmentations : nombre de triangles et d'arêtes, temps de calcul, nombre de plans, de sphères et de cylindres. $(^*)$: le modèle d'origine a été modifié.

6.3 Remaillage

Le remaillage a comme objectif de prendre un maillage surfacique en entrée et d'en produire une version améliorée approximant la surface du maillage d'entrée le plus fidèlement que possible. Les imperfections d'un maillage à supprimer sont souvent dues à ses origines :

- la numérisation d'une objet 3D réel par un scanner ;
- la création, à l'aide d'un algorithme de vision, à partir d'une ou plusieurs images ;
- la modélisation avec des logiciels dédiés pour la production de pièces mécaniques ou pour les films d'animation ou pour les jeux vidéos.

Quelque soit l'origine d'un maillage, sa structure n'est pas toujours adaptée aux besoins de l'application en question, par exemple la compression, la visualisation, le rendu, le calcul numérique etc. Dans ce contexte, la qualité d'un maillage dépend d'une multitude de facteurs. Nous donnerons ici une liste courte, pour une discussion sur ce sujet nous renvoyons le lecteur à [Vid11, She02, AAGU07] :

- l'échantillonnage de la surface par rapport aux degrés locaux de détails ; la conservation de ses caractéristiques comme les hautes fréquences et les arêtes vives ;
- la régularité de la connectivité ;
- la qualité des triangles. Pour la plupart des applications, des triangles équilatéraux (ou proches) sont souhaités.

Ici nous reprenons la classification de l'état de l'art fait par Vincent Vidal dans [Vid11] en 4 types.

La première famille de méthodes est basée sur le ré-échantillonnage explicite, c.à.d. le contrôle de l'échantillonnage permettant de contrôler à la fois la densité du maillage et sa qualité. Elles procèdent par une suite séquentielle d'opérations, comme i) l'insertion d'un point à la fois, toute en gardant une triangulation de Delaunay (pour chaque triangle, le cercle circonscrit ne contient pas d'autres sommets) [PW04, DLR05] ; ii) la suppression de points et re-triangulation du trou produit [SZL92] ; iii) la propagation de fronts, c.à.d. la création d'un pavage à partir d'un triangle initial en plaçant les triangles aux bords de manière optimale selon certains critères [AFSW03, DKG05]. Un problème difficile à résoudre concerne la rencontre de deux fronts, où le pavage est forcement sous-optimal nécessitant des post-traitements couteux.

Le deuxième groupe passe par une paramétrisation, c.à.d. une bijection entre le maillage et un plan paramétrique 2D. Le problème de remaillage et résolu plus facilement et rapidement dans l'espace paramétrique, ensuite la triangulation est projetée sur le maillage [AMD02, AdVDI05]. Par contre, une paramétrisation introduit nécessairement une distorsion, ce qui diminue la qualité du résultat. Des problèmes son également posés par les maillages fermés et les genres > 0. Souvent cela est résolu par des paramétrisations locales [SG03], ce qui pose des problèmes aux frontières des patches.

La troisième famille optimise le maillage en repositionnant ses sommets, soit de manière individuelle, soit globalement. La connectivité fixe de ces méthodes va restreindre la quantité de modifications possibles. Pour cette raisons, ces algorithmes sont souvent alternés avec des étapes de modification de connectivité (voir le 4^e groupe ci-dessous). Le positionnement local positionne les sommets individuellement en fonction de leurs voisins. Un des opérateurs les plus connus est le lissage Laplacien, qui déplace les sommets en direction du barycentre du one-ring [Tau95]. Il est très répandu malgré ses défaut connus, notamment l'introduction de distorsions géométriques, un manque de contrôle sur le repositionnement global, des instabilités numériques et une convergence lente. D'autres opérateurs locaux sont le lissage basé angle, égalisant les angles des triangles [SG04], l'égalisation des aires (également introduit dans [SG04]), et l'opérateur de flot de courbure conçu pour le débruitage [DMSB99].

Toujours dans la troisième famille, les méthodes globales de repositionnement de sommets passent par la minimisation d'une fonction d'énergie globale. Contrairement à notre solution, ce problème est résolu directement dans l'espace continu des positions des sommets. Si on considère des critères simples similaires aux opérateurs locaux ci-dessus, le problème peut se formaliser comme un système d'équations linéaires qui peut être résolu au sens des moindres carrés [LTJW07, WHG08]. Le défaut de ces méthodes est un lissage souvent trop important. D'autres méthodes essaient de produire un diagramme de Voronoï Centroidal (CVT), c.à.d. un diagramme de Voronoï dans lequel les générateurs des cellules sont aux barycentres des cellules [SAG03, CSAD04, VCP08]. Souvent cela est fait par des méthodes itératives, e.g. l'alternance entre le calcul du diagramme de Voronoï et le déplacement des centres vers les barycentres. Dans [CSAD04], cette étape est faite dans le cadre d'une approximation de la surface par plans — voir aussi la section 6.2 sur la décomposition de maillages.

Les méthodes du 4^e groupe optimisent la connectivité d'un maillage. Elles sont souvent combinées avec les méthodes de positionnement de sommets afin de s'approcher d'une connectivité optimale, ç.à.d. une valence de 6 pour chaque sommet, et une valence de 4 pour les sommets aux bords. Ces méthodes peuvent néanmoins aussi améliorer la qualité du maillage. Elles emploient des opérations locales de changements de connectivité, telles que :

- le basculement d'arête, qui transforme une paire de triangles voisins séparée par une arête en une autre paire impliquant les mêmes sommets ;
- la contraction d'arête, qui supprime une paire de triangles voisins et une arête en fusionnant deux paires d'arêtes ;
- la découpe d'arête, qui transforme une paire de triangles voisins en un ensemble de 4 triangles.

La question centrale est l'ordre dans lequel ces opérations sont à exécuter sur le maillage. A notre connaissance, aucune solution optimale et globale n'a été trouvée pour ce problème combinatoire difficile. La plupart des méthodes emploient des solutions gloutonnes basées sur une file de priorité, comme dans [HDD+93], ou d'autres stratégies gloutonnes [SG03], ou ils effectuent les opérations

aléatoirement [AMD02]. Une tentative de globalité recourt au recuit simulé [YGZW07], qui est douloureusement lent — voir aussi le chapitre 3, section 3.4.3.

Modélisation

Nos travaux sur le remaillage [VWLD09, VWD12a] ont les objectifs suivants :

- l'amélioration de la qualité des triangles tout en gardant une excellente approximation de la surface ;
- la préservation des arêtes caractéristiques et des coins du maillage ;
- l'applicabilité à tout maillage 2-variété, comprenant les maillages de genres arbitraires ;
- la création de maillages semi-réguliers n'est pas notre objectif.

Les approches de l'état de l'art, décrites ci-dessus, peuvent être séparées de manière alternative en deux groupes : le premier emploie des opérations locales dont l'ordre est choisi de manière gloutonne. Cela permet un bon contrôle de l'erreur locale souvent avec des garanties de convergence. Le deuxième groupe optimise une fonction globale, ce qui les empêche de satisfaire des contraintes dures sur les modifications locales. Notre point de départ est le suivant. Une approche globale est souhaitable, car cela permet d'obtenir une meilleur cohérence de la solution. Il est également préférable de procéder par un suite d'opérations locales afin de contrôler l'erreur introduite. Notre idée est donc de passer par une optimisation effectuée sur un ensemble d'opérations locales.

Commençons avec les aspects globaux. Notre approche interprète la tâche comme un problème d'optimisation globale. Il s'agit de trouver une configuration optimale pour le nombre de sommets, les positions des sommets et pour la structure du maillage. L'optimalité est mesurée par rapport à la qualité du maillage (comme définie au début de la section) et par rapport à la fidélité de la surface approximée, tout en satisfaisant des contraintes sur la topologie et sur le type de maillage. Le problème est donc de nature mixte : l'optimisation des positions des sommets est un problème continu dans $\mathbb{R}^3$, et l'optimisation de la structure du graphe est un problème combinatoire. Un autre verrou correspond à la dépendance entre la densité du maillage et la qualité. L'optimisation du premier problème aura un effet sur le nombre de variables du deuxième problème.

Nous avons résolu ce problème en discrétisant le problème continu et en séparant l'optimisation des positions d'un coté, et de la connectivité et de la densité de l'autre coté. La figure 6.9 donne une illustration de cette approche. La discrétisation propose, de manière intelligente, des candidats pour des nouvelles positions pour les sommets, transformant ainsi le positionnement global de tous les sommets du maillage en un problème combinatoire. Les étapes de repositionnement sont alternées avec des étapes d'optimisation de connectivité, qui optimisent également la densité du maillage. A quelques exceptions près, les deux étapes sont basées sur la même fonction d'énergie globale, ce qui permet de garder une cohérence entre les étapes.

Deux types de variables sur les sommets s du maillage interviennent dans la fonction d'énergie. Les variables cachées x_s peuvent prendre des valeurs dans $\{0, 1\}$, où $x_s{=}0$ signifiera que le sommet s gardera sa position actuelle, tandis que $x_s{=}1$ signifiera que la nouvelle position proposée sera retenue pour le sommet s. Les variables y_s correspondent aux positions des sommets s dans le maillage d'origine. Le reste de la notation est identique à la notation utilisée dans la section 6.2

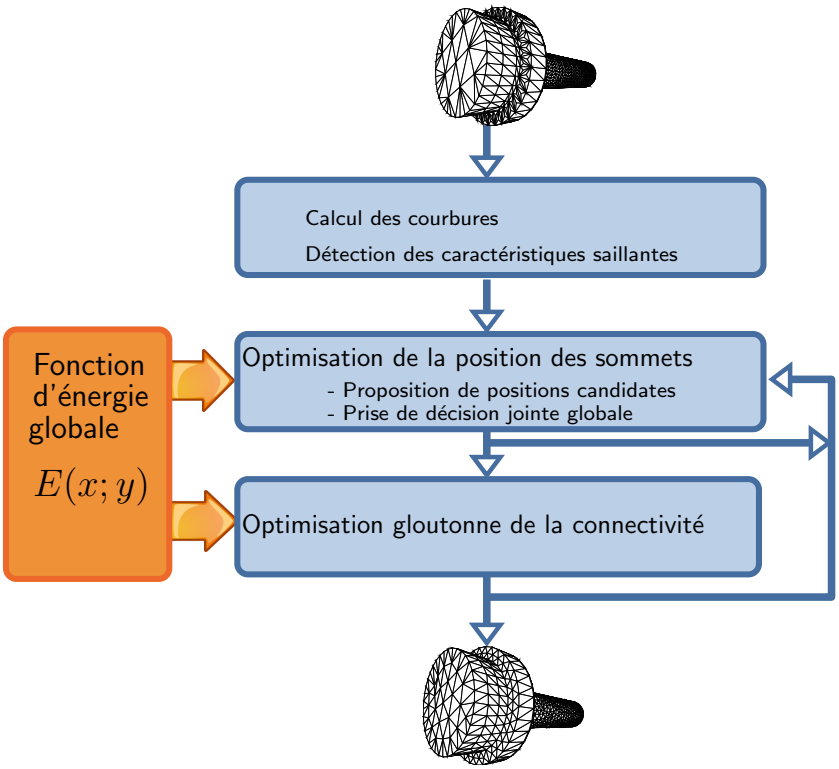


Figure 6.9 – Un schéma illustrant la méthode de remaillage.

pour la décomposition de maillages. La fonction d'énergie optimisée est la suivante :

$$\begin{aligned} E(x;y) &= \lambda_{fo} \sum_{\{s,r,q\}\in\mathcal{S}^{(t)}} E_{forme}(x_s,x_r,x_q) \\ &+ \lambda_{fi} \sum_{\{s,r,q\}\in\mathcal{S}^{(t)}} E_{fidelite}(x_s,x_r,x_q;y) \\ &+ \lambda_{v} \sum_{s\in\mathcal{S}^{(s)}} E_{valence}(x_s) \\ &+ \lambda_{b} \sum_{s\in\mathcal{S}^{(s)}} E_{budget}(x_s) \end{aligned} \tag{6.11}$$

Ici, les différents λ_x sont les poids associés aux différents types de termes. Cette notation indique uniquement la dépendance aux positions des sommets. Cependant, la connectivité du graphe fait partie des « variables » à optimiser, et il va sans dire que l'énergie en dépend. Cela se voit dans la définition des termes de budget qui contrôlent la densité du maillage. Ils sont tout simplement constant, ce qui rend l'énergie linéaire par rapport au nombre de sommets :

$$E_{budget}(x_s) = 1 \tag{6.12}$$

La densité peut donc être contrôlée par le paramètre associé λ_b. Selon le paramétrage de notre méthode, le maillage produit a une densité similaire ou moins élevée. Notre méthode peut donc être également utilisée pour la simplification de maillages. Ces termes interviennent surtout lors de

l'étape d'optimisation de connectivité. Ils peuvent être ignorés lors de l'optimisation des positions des sommets.

Les termes de forme favorisent les triangles équilatéraux [PB03] :

$$E_{forme}(x_s, x_r, x_q) = \frac{R(x_s, x_r, x_q)}{\min(||x_s - x_r||, ||x_s - x_q||, ||x_r - x_q||)} \tag{6.13}$$

où $R(.)$ est le rayon du cercle circonscrit et $||.||$ est la norme L_2. Pour des raisons des simplicité, dans (6.13) nous avons abusivement utilisé les variables discrètes x_s pour décrire des positions et non pas pour des choix.

Contrairement à l'énergie d'un champ classique, par exemple pour la restauration d'images (voir aussi le chapitre 5), les termes de fidélité ne sont pas définis par sommet mais par triangle. L'objectif est d'assurer une approximation fidèle de la surface d'origine. Ils sont définis comme le volume de la projection orthogonale du triangle remaillé sur la surface d'origine. Pour des raisons de complexité, durant la phase de repositionnement des sommets cette erreur est remplacée par une distance calculée par sommet :

$$E^*_{fidelite}(x_s, x_r, x_q; y) = F(x_s, y) + F(x_r, y) + F(x_q, y) \tag{6.14}$$

où $F(x_s, y)$ est la distance carrée entre x_s et le maillage d'origine.

Les termes de valence favorisent des sommets avec des degrés (nombre des voisins) optimaux :

$$E_{valence}(x_s) = \left(\; d(x_s) - d_{opt}(x_s) \; \right)^2 \tag{6.15}$$

où $d(.)$ est le degré d'un sommet et d_{opt} est son degré optimal, c.à.d. 6 dans le cas général et 4 aux bords.

Le graphe de dépendances pour ce modèle est donné dans la figure 6.10 pour une sommet et ses voisins directs. Les variables à optimiser x_s correspondent aux sommets du maillage — nous ignorons ici l'optimisation de la connectivité. Les termes d'ordre supérieur sont définis sur les triangles, donc sur les cliques de taille 3. Chaque variable cachée x_s, associée à un sommet du maillage remaillé, dépend potentiellement de toutes les observations du modèle, correspondant aux sommets du maillage d'origine. Notre méthode n'est pas probabiliste ; néanmoins, à cause de cette dépendance, un modèle graphique probabiliste dont la fonction d'énergie serait donnée dans l'équation (6.11), serait du type CRF — voir aussi le chapitre 3, section 3.3.

En pratique, à chaque instant, pour chaque sommet du maillage remaillé, un lien vers le sommet le plus proche du maillage d'origine est gardé. Ce sommet et son voisinage sont utilisés pour calculer l'erreur intervenant dans le terme de fidélité.

Propositions de positions

A chaque itération de l'algorithme, une nouvelle proposition de position est calculée pour chaque sommet du maillage. Afin de contrôler l'erreur, d'éviter des replis du maillage, et pour limiter l'introduction de bruit de hautes fréquences, ces positions sont restreint à une sphère de liberté décroissante dans le temps, similaire à la température de l'algorithme recuit simulé. Plusieurs candidats sont possibles pour chaque sommet, calculés par des opérateurs différents :

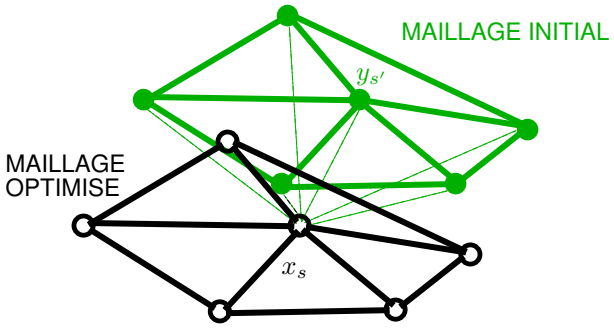


Figure 6.10 – Le graphe de dépendances pour l'optimisation des positions des sommets.

— Le lissage basé angle ;
— Le lissage Laplacien ;
— Un pas dans la direction du gradient de l'énergie ;
— Un échantillonnage aléatoire.

Toutes ses candidats sont évalués dans cet ordre, et le premier qui tombe dans la sphère de liberté est proposé comme alternative à la position actuelle. Les nouvelles positions proposées respectent également les caractéristiques du maillage, préalablement détectées. Ainsi, un sommet sur une arête vive ne pourra bouger uniquement le long de l'arête ; un sommet sur un coin ne pourra pas changer de position.

Décision globale sur les positions

La fonction d'énergie (6.11) n'est pas sous-modulaire, car cette condition n'est pas nécessairement vérifiée pour les termes E_{forme} pour tous les triangles. Inspiré par le calcul de flot optique dans les vidéos introduit dans [LRR08], notre algorithme calcule la meilleure combinaison entre l'ensemble des positions actuelles et l'ensemble des positions proposées à l'aide d'un algorithme d'optimisation discrète de la famille des *graph cuts*, soit la variante QPBO [KR07]. La qualité de la solution dépend du nombre de termes non sous-modulaires. Si ce nombre n'est pas zéro, certaines étiquettes ne seront pas trouvées par l'algorithme. Nous avons constaté dans nos expériences, que QPBO était capable de trouver une solution pour 99% des étiquettes, pour lesquelles la solution était donc optimale.

Optimisation de la connectivité

L'optimisation de la connectivité intervient moins fréquemment que les opérations de positionnement des sommets, soit toutes les 5 itérations. Comme pour les méthodes de l'état de l'art, cette optimisation est basée sur les opérations locales connues : basculement d'arêtes, découpes d'arêtes, contractions d'arêtes (voir la page 157). Également comme dans les travaux existants, leur ordonnancement est glouton et basé sur des files de priorité [HDD+93]. Par contre, à la différence des méthodes connues, la priorité d'une opération est directement liée à la différence d'énergie globale (6.11).

Certaines opérations changent le nombre de termes dans la fonction d'énergie (6.11). Cela concerne le terme de budget E_{budget}, prévu pour ça, mais également les termes de forme, de fidélité

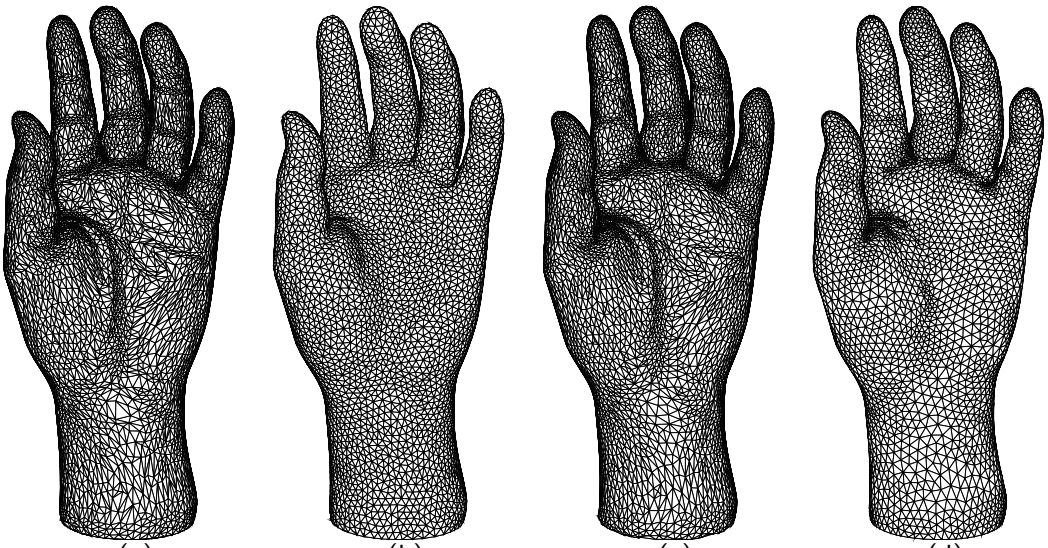


Figure 6.11 – Comparaisons entre (a) le modèle hand original, (b) Valette et al. [VCP08] (c) Liu et al. [LTJW07] et (d) notre méthode.

et de valence. Le paramètre de budget λ_b a donc un double role : i) compenser la suppression des termes de fidélité, de forme et de valence, pour que la solution optimale ne soit pas égale au maillage vide ; ii) contrôler la densité souhaitée du maillage.

Résultats

Nous avons comparés notre méthode avec plusieurs autres méthodes de l'état de l'art :

- Deux méthodes basées sur la CVT par clustering : Valette et al. [VCP08] et Surazhky, Alliez et Gotsman [SAG03]
- Une méthode basée sur des opérations locales : repositionnement en égalisant les aires des triangles, et changement de connectivité, proposée par Surazhky et Gostman [SG03] ;
- une méthode basée sur le positionnement global des sommets par moindres carrés, proposée par Liu et al. [LTJW07] ;

Les résultats sont donnés dans la table 6.2. Les maillages obtenus par notre méthode sont généralement les meilleurs, selon les angles minimaux et maximaux moyens. Nous pouvons constater que cette qualité a été obtenue malgré une très bonne approximation de la surface d'origine. Les figures 6.11 et 6.13 montrent des objets remaillés par des méthodes différentes. Nous pouvons constater que les détails sont particulièrement bien conservés, comme par exemple l'œil du triceratops dans la figure 6.13, surtout par rapport à la méthode concurrente. L'application aux maillages de 2-variété de genre élevé est possible, comme on peut le voir dans la figure 6.14. La simplification de maillages est possible par une simple adaptation des paramètres de la méthode, comme illustré dans la figure 6.12.

Modèle	#v	Irreg (%)	Amin (deg)	Amax (deg)	Er_{Haus} (10^{-3})	Er_{RMS} (10^{-3})	Durée (sec)
Fandisk (init)	6495	20	43.4	86.1	-	-	-
Fandisk [LTJW07]	6495	20	44.7	82.0	3.3	0.8	n.c.
Fandisk (notre)	5905	**12**	**49.0**	**75.9**	**1.6**	**0.03**	232
Cow (init)	2904	53	30.2	93.7	-	-	-
Cow [LTJW07]	2904	53	35.1	88.2	**5.3**	0.9	n.c.
Cow (notre)	2695	**39**	**41.0**	**81.0**	5.5	**0.5**	59
Shark (init)	2560	32	20.8	97.4	-	-	-
Shark [LTJW07]	2560	32	26.2	107.5	**3.0**	**0.3**	n.c.
Shark [SAG03]	2560	**31**	**50.6**	**71.1**	6.8	0.8	n.c.
Shark (notre)	1719	47	36.2	84.8	4.0	0.6	42
Hand (init)	7950	58	32.4	94.1	-	-	-
Hand [LTJW07]	7950	58	34.3	92.2	8.8	0.4	n.c.
Hand [VCP08]	6802	45	46.1	77.5	2.6	**0.2**	9
Hand (notre)	5847	**33**	**50.2**	**72.3**	**1.7**	**0.2**	193
Bimba (init)	8857	62	34.2	92.8	-	-	-
Bimba [LTJW07]	8857	62	38.1	87.0	4.9	0.5	n.c.
Bimba [SAG03]	8857	**20**	**53.6**	**67.6**	6.0	0.5	n.c.
Bimba [VCP08]	8143	48	45.2	78.1	6.0	0.4	10
Bimba (notre)	7986	41	47.6	75.3	**3.0**	**0.2**	232
Egea (init)	8268	75	34.7	93.5	-	-	-
Egea [LTJW07]	8268	75	38.2	88.3	**2.6**	**0.2**	n.c.
Egea [SG03]	8705	**7**	**52.4**	**69.1**	2.7	**0.2**	15
Egea (notre)	7783	43	48.8	74.1	**2.6**	**0.2**	236
Triceratops (init)	2832	59	29.6	95.5	-	-	-
Triceratops [SG03]	2758	**13**	**42.2**	82.5	8.4	1.1	12
Triceratops (notre)	2412	44	41.5	**81.0**	**3.6**	**0.5**	55

Table 6.2 – Comparaison avec [LTJW07], [VCP08], [SAG03] et [SG03]. Les mesures sont : nombre de sommets, pourcentage de sommets irréguliers, angle minimal moyen, angle maximal moyen, distance de Hausdorff, maximum entre les 2 distances RMS mesurées par Metro normalisées par la diagonale de la boite englobante, et le temps d'exécution. Les temps affichés pour [SG03] ont été calculés sur un Pentium 4 PC (2.4 GHz) avec 512 de RAM, tandis que les autres l'ont été sur un Intel Core 2 Duo P8400 (2.26 GHz) avec 4 Go de RAM.

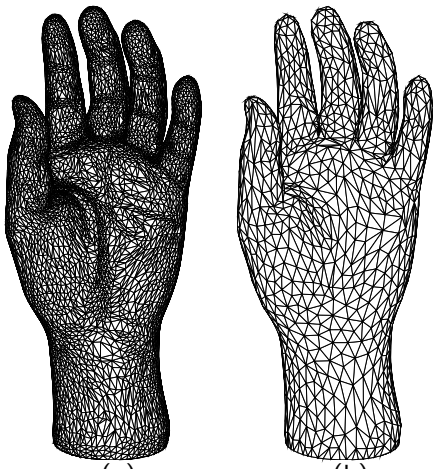


Figure 6.12 – Simplification : (a) modèle original ; (b) modèle simplifié (1518 sommets).

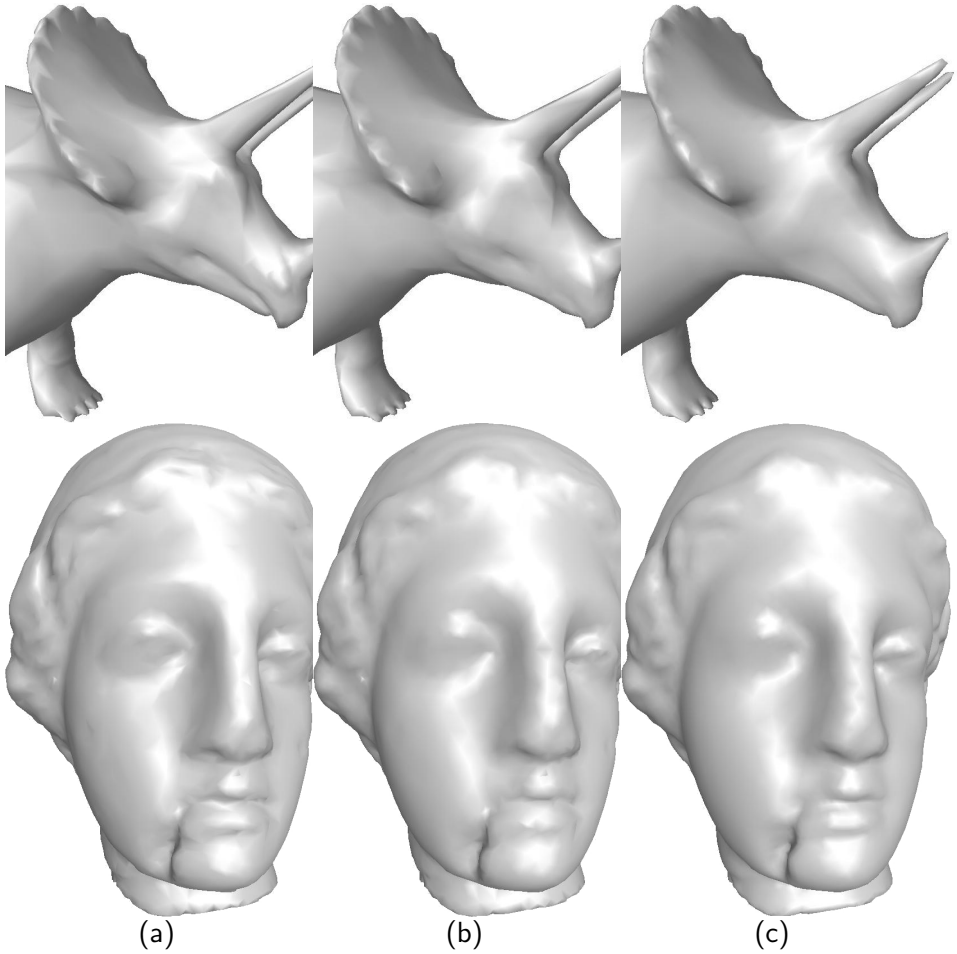
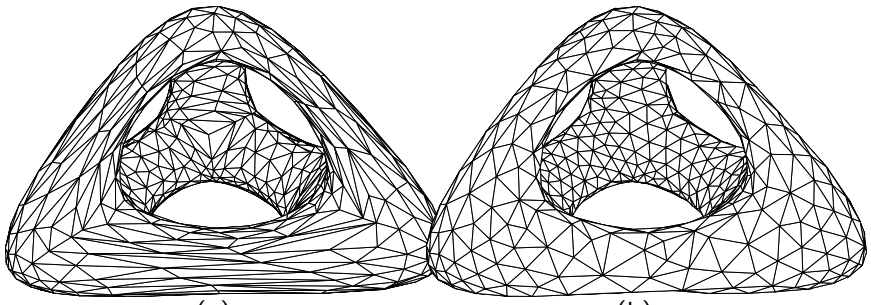


Figure 6.13 – Comparaison : (a) modèles d'origine ; (b) remaillé par notre méthode ; (c) remaillé par [SG03].



(a) (b)

Figure 6.14 – Modèle de genre 3 : (a) modèle d'origine ; (b) version remaillée.

6.4 Un opérateur de zoom pour les très grandes images par déformation de maillages 2D

Dans cette section nous décrivons nos travaux [LJW+10] sur une application qui, *a priori*, ne serait pas liée à la thématique de ce chapitre. Il s'agit de la problématique de diminuer la taille d'une image afin de pouvoir la visualiser sur un écran de taille plus petite, par exemple pour passer d'un grand écran HD vers le petit écran d'un téléphone portable. La solution classique, l'échantillonnage uniforme, est peu adaptée à ce problème si les proportions des deux écrans sont différentes, ou si l'image contient des grandes parties sans contenu « intéressant ». Si un manque de fidélité n'est pas un problème dans le contexte applicatif, si l'image peut donc être légèrement changée lors de la visualisation, une meilleure solution est possible : le recadrage avec prise en compte du contenu. Il s'agit d'un processus de changement de taille non-uniforme, qui diminue plus fortement les parties de l'image considérées comme moins intéressantes. Ces méthodes dépendent donc d'un détecteur de zones de « saillance », ou d' « intérêt » dans l'image. La discussion de ce qui est considéré comme saillant est hors de portée de ce mémoire. Remarquons uniquement que la plupart de techniques définissent la saillance à partir d'une mesure d'énergie du signal, par exemple dans [AS07, RSA08]. Une technique complémentaire récemment proposée estime la saillance à partir du comportement de zoom d'un ensemble d'utilisateurs [CCO+10].

Les premiers travaux sur ce sujet étaient basés sur le *seam carving*, le calcul de lignes contiguës traversant l'image sur des chemins de saillance minimale [AS07, RSA08]. Rapidement les méthodes basées sur la déformation de maillages 2D ont été prouvées comme étant meilleures [WTSL08, GLS+09], ce qui nous fait revenir sur la thématique de ce chapitre. Un maillage est plaqué sur l'image et déformé, en tenant compte de la taille souhaitée et de la carte de saillance préalablement calculée. L'image recadrée est rendue avec des techniques de plaquage de textures.

Nos travaux sont basées sur la méthode de Wang et al. [WTSL08], qui déforme le maillage en minimisant une fonction d'énergie globale sur les sommets $\mathbf{v}$:

$$E_{forme}(\mathbf{v}) = \sum_{\mathbf{t}\in\mathcal{S}^{(t)}} w_t \sum_{(ij)\in\mathcal{E}(\mathbf{t})} \left\|(\mathbf{v}'_i - \mathbf{v}'_j) - F_{\mathbf{t}}(\mathbf{v}_i - \mathbf{v}_j)\right\|^2 \tag{6.16}$$

Ici $\mathbf{t}$ sont les triangles du maillage et $\mathcal{E}(\mathbf{t})$ sont les arêtes du triangle $\mathbf{t}$. Les w_t sont les poids associés à chaque triangle ; ils sont liés à la carte de saillance. Les $F_{\mathbf{t}}$ sont les facteurs d'échelle agrandissant un triangle $\mathbf{t}$ de manière uniforme — ces facteurs sont eux même dépendants des positions des sommets, une dépendance omise dans la notation. L'énergie mesure donc l'écart entre

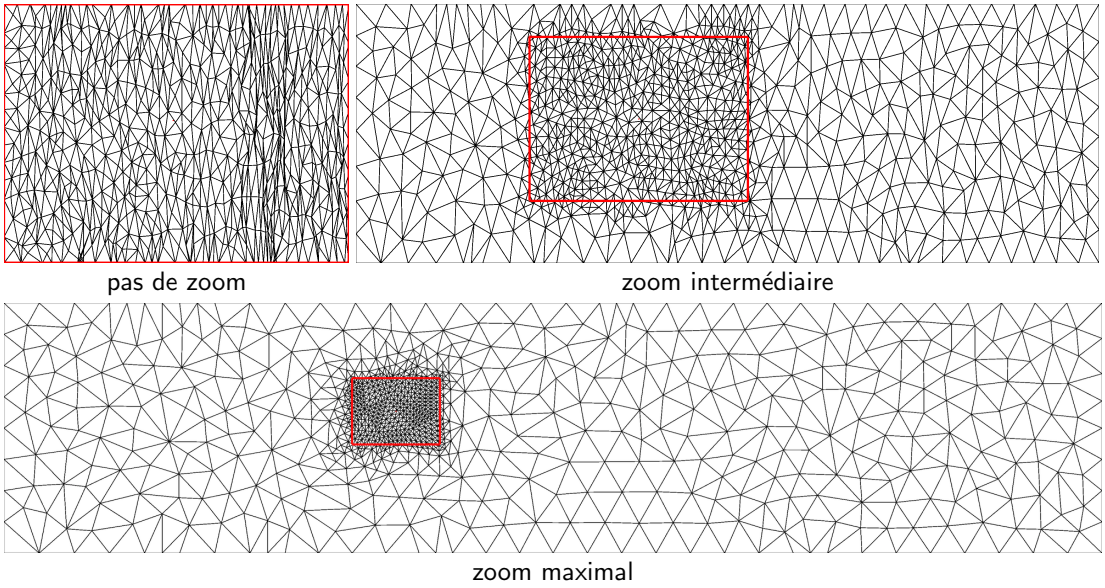


Figure 6.15 – Le maillage à trois différents niveaux de zoom. Le rectangle rouge correspond à la zone de navigation.

un agrandissement uniforme et l'agrandissement réel. L'énergie est minimisée avec un algorithme itératif alternant entre l'optimisation des $F_{\mathbf{t}}$ et l'optimisation des $\mathbf{v}$. Cela rend l'énergie quadratique et donc minimisable en résolvant un système linéaire creux. L'énergie peut être étendue en ajoutant d'autres contraintes, par exemple sur la préservation de lignes. Cette méthode peut d'ailleurs être liée à une méthode très similaire pour le remaillage surfacique 3D [LTJW07] brièvement discuté dans la section 6.3, et dont les résultats sur les maillages 3D surfaciques sont illustrés dans la figure 6.11c.

Nos contributions se situent dans le contexte de la navigation dans les très grandes images — jusqu'à 10^9 pixels et plus — sur des petits écrans et avec prise en compte du contenu de l'image. La solution ad hoc, l'application d'un algorithme de recadrage non-uniforme pour obtenir des proportions souhaitées, suivi par le rognage de l'image, se heurterait à plusieurs problèmes :

— les images sont trop grandes pour un calcul interactif ;
— les distorsions générées par cette approche sont trop grandes ; surtout, au niveau de zoom maximal, des distorsions seront toujours visibles, ce qui n'est pas souhaité.

Notre méthode proposée se sert d'un maillage adaptatif dont la densité local dépend du niveau de zoom et de la zone de zoom choisi, comme le montre la figure 6.15. Un maillage très fin est nécessaire pour la zone inspectée pour minimiser les distorsions introduites. Un maillage très grossier est nécessaire pour le reste de l'image pour garantir une réponse interactive du système. En changeant le niveau de zoom, la taille et les proportions de la zone inspectée sont adaptés.

La structure de données utilisée pour ce maillage est une forêt, où chaque racine correspond à un triangle du maillage le plus grossier. Un schéma de subdivision va générer un maillage de plus en plus fin jusqu'à une limite dépendant du nombre de pixels couverts par un triangle. Les triangles générés par un triangle de base correspondent aux descendants dans la hiérarchie. Pour un certain niveau de zoom demandé, une coupe est créée dans cette structure, produisant le maillage adaptatif (avec une densité hétérogène) illustré dans la figure 6.15. Pour éviter des artefacts entre deux frames

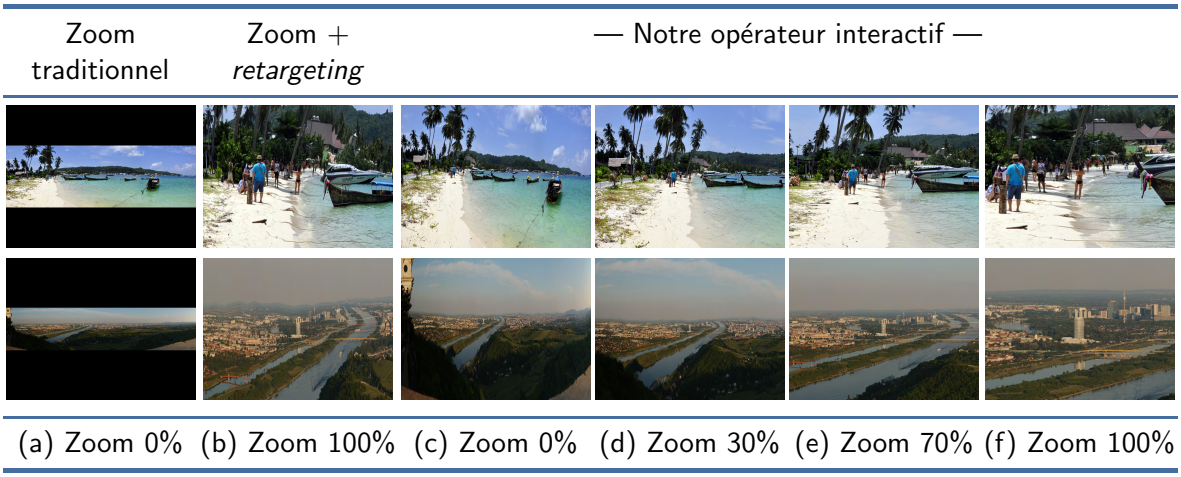


Figure 6.16 – Comparaison visuelle : (a) zoom classique ; (b) Recadrage non-uniforme + Zoom ; (c-f) Notre opérateur. Veuillez zoomer pour voir plus de détails.

d'un mouvement de zoom, une interpolation de type *geomorphing* [Hop97] est appliquée entre les deux maillages respectifs.

La figure 6.16 montre une comparaison entre les résultats obtenus avec le nouvel opérateur de zoom et deux méthodes concurrentes : i) le zoom classique par ré-échantillonnage ; ii) l'opérateur de recadrage non-uniforme suivi par un zoom classique. Les écrans virtuels de visualisation étaient de taille 640×480 pixels, sauf pour l'image de la dernière ligne pour laquelle la taille était de 400×300 pixels. Les résultats pour des différents niveaux de zoom partagent le même centre de vue. Le zoom traditionnel montre peu d'informations dans le niveau de zoom le plus faible, et selon la différence de proportions entre l'image et l'écran, une grande partie de l'écran peut être gaspillée par des barres noires. L'opérateur employant séquentiellement recadrage et zoom montre des distorsions géométriques même dans le niveau de zoom le plus élevé. Notre opérateur utilise toujours tout l'espace disponible et la distorsion induite diminue au fur et à mesure que le niveau de zoom est augmenté.

Le temps de réponse du système dépend de la taille que couvrent les triangles. Il varie de <100ms pour 500 pixels par triangle vers 200ms pour 200 pixels sur un Intel Xeon de 2.6Ghz et 4GO de RAM.

6.5 Conclusion et perspectives

Dans ce chapitre nous avons présenté nos travaux sur l'analyse de maillages 3D surfaciques et de maillages 2D à l'aide de modèles graphiques. Nous avons montré que la recherche de solutions globalement cohérentes peut être une approche efficace donnant des résultats de très bonne qualité.

Nous mentionnons ici très brièvement quelques perspectives pour ce chapitre. Nous renvoyons le lecteur au chapitre 7 pour plus de détails sur ces idées et sur une description plus générale des perspectives de nos travaux.

Décomposition + remaillage La décomposition de maillages, décrite dans la section 6.2 pourrait être une étape de pré-traitement intéressante pour notre algorithme de remaillage, décrit

dans la section 6.3. Il existe des algorithmes de remaillage munis d'une étape préalable de décomposition, par exemple VSA [CSAD04]. Cependant, au lieu de procéder par une triangulation explicite des proxies détectés par l'étape de segmentation, comme cela est fait dans [CSAD04], nous proposons que l'algorithme de remaillage utilise le résultat de la décomposition afin de guider la génération des candidats pour les positions des sommets. Comparé à [CSAD04], cela rendrait le résultat du remaillage moins dépendant des erreurs de l'étape de décomposition.

Modèles hiérarchiques La conception de modèles hiérarchiques pour les objets 3D, courant en segmentation d'images, pourrait être bénéfique pour la segmentation de maillages et permettre de traiter des maillages de plusieurs millions de sommets.

Optimisation La plupart des problèmes étudiés demandent la minimisation de fonctions contenant des termes non sous-modulaires. L'utilisation d'algorithmes classiques est difficile dans ce cas, la conception d'algorithmes dédiés pour chaque problème est une piste prometteuse.

Méthodes variationnelles nous envisageons à l'avenir l'étude d'une décomposition de maillages en primitives géométriques basée sur la variation totale.

Images RGB+D Les caméras de profondeur récemment introduites au marché (MS Kinect & Co) livrent des images en couleur et en profondeur. Certains applications, comme la reconnaissance d'actions, pourraient profiter d'un traitement directement en 3D.

Chapitre 7

Conclusion générale et perspectives

Dans ce mémoire nous avons présenté les travaux que nous avons menés entre 2005 et 2012 et nous avons déjà brièvement mentionné quelques perspectives dans les chapitres précédents. Dans ce dernier chapitre nous donnerons une conclusion générale et nous décrirons en détails les recherches que nous prévoyons pour l'avenir. Nous avons tenté de présenter les perspectives sur un horizon variable : certains travaux sont actuellement en cours ; d'autres travaux sont en cours de discussion avec nos doctorants et nos collaborateurs, ou leur détails font actuellement objet d'une réflexion personnelle ; enfin, d'autres travaux en perspective ici discutés sont des pistes que nous envisageons en long terme, en jugeant que la recherche se tournera probablement dans cette direction.

D'un point de vu applicatif, les travaux décrits dans ce mémoire se sont largement concentrés sur deux aspects, à savoir les problèmes de détection et de reconnaissance, ainsi que la segmentation et la restauration. Les premiers travaux traitaient les images de documents dans un sens large. Ces dernières années, nous nous sommes tournés plus vers le traitement de vidéos issues de différentes sources, par exemple la surveillance, les émissions de télévision, et la robotique mobile. Le traitement de modèles 3D est un autre axe sur lequel nous avons travaillé. Dans l'avenir nous aimerions d'ailleurs combiner les deux dernières thématiques en traitant des vidéos par approches 3D.

D'un point de vu plus théorique, le fil rouge de nos travaux est la recherche de solutions cohérentes, surtout par modèles structurés. Dans nos travaux, les structures (telles que les graphes) peuvent servir comme support de calcul, par exemple dans le cas d'un graphe de dépendances, ou comme représentation d'une entité même, par exemple d'un objet ou d'une action. Cette distinction peut également délimiter deux sous-thématiques de nos contributions : i) la modélisation par modèles graphiques probabilistes, ou de manière alternative par fonction d'énergie globale non-probabiliste ; ii) la modélisation d'objets et d'actions par modèles semi-structurés, maximisant invariance et pouvoir de discrimination.

7.1 La vision par ordinateur dans les 5 à 10 ans à venir

La vision par ordinateur a fait des progrès significatifs durant les dernières 10 ans. Nous sommes passés d'une thématique abordée dans les laboratoires de recherche, et appliquée en industrie dans des situations très maitrisées, à des applications de la vie quotidienne. Les systèmes de vision sont maintenant capables de reconnaitre des gestes humains en temps réel, sur un matériel informatique conçu en 2006, et dans des situations non maitrisées, à savoir les pièces mal éclairées de 10 millions

de foyer dans le monde [SFC+11]. La détection de visages a été intégrée dans des appareils photos et dans des téléphones mobiles, et la reconnaissance faciale a été intégrée sur des sites web comme *Facebook*.

Malgré ces avancées, ils reste un long chemin à parcourir pour les chercheurs dans ce domaine. Nous sommes encore loin d'un système de vision capable de détecter et de reconnaitre des objets à partir d'un ensemble important de modèles, dans des poses arbitraires et dans des conditions difficiles, pour citer un seul exemple. A notre avis, plusieurs pistes viendront à notre aide. La modélisation du contexte nous permettra d'intégrer un grand nombre d'informations supplémentaires pour dépasser le fossé sémantique séparant le signal de bas niveau des informations de haut niveau liées à la sémantique — voir les perspectives décrites dans la section 7.2.3. Par contre, les représentations de ces relations sont de nature structurée, ce qui rend difficile leur intégration dans des algorithmes d'apprentissage. Les modèles structurés et leur apprentissage restent parmi les thématiques que comptons poursuivre — voir la section 7.3.

De manière générale, les représentations structurées gagneront sans doute en importance dans l'avenir, tant pour une modélisation du contexte, que pour la structuration de problèmes de bas niveau. La modélisation de ce genre de relations est traditionnellement faite à l'aide de modèles pour lesquels l'inférence nécessite la résolution de problèmes complexes, souvent combinatoires, aboutissant à des algorithmes de complexité de calcul importante. D'un autre coté, nous savons que ces problèmes sont résolus en temps réel par le système visuel humain, capable de réagir rapidement et de s'adapter à des situations inconnues. Une piste prometteuse, inspirée de nos connaissances sur le système visuel humain, est de procéder par modélisation hiérarchique et en alternant des parcours ascendants et descendants. Ces approches ont donné des excellents résultats sur des tâches impliquant des informations de bas niveau — voir les modèles hiérarchiques pour la segmentation présentés dans le chapitre 5. L'extension à des raisonnements de plus haut niveau nécessite la modélisation et l'apprentissage d'un modèle hiérarchique capable de représenter, sur des échelles différentes, des données complexes, structurées, comprenant des interactions complexes — voir les sections 7.2.2 et 7.4.2.

L'apprentissage jouera un rôle primordial dans ce contexte. Nous pouvons mentionner les modèles profonds, tels que les *Deep Belief Network* ou les réseaux convolutionnels, dont l'approche hiérarchique globale permet d'intégrer l'extraction de caractéristiques et la prise des décisions dans un seul modèle. Leur complexité (dans le sens du nombre de paramètres, donc la complexité du modèle de prédiction) est telle qu'ils sont habituellement appris de manière non-supervisée, couche par couche, à l'exception des dernières couches. La résolution de problèmes plus complexes nécessite l'augmentation de la complexité du modèle de prédication, donc du nombre de paramètres, ce qui, très rapidement, entraine un besoin prohibitif de quantité de données d'apprentissage. La structuration du modèle permet de diminuer ou de limiter le problème de *overfitting*, soit par la structuration du modèle même, comme dans le cas des réseaux convolutionnels, soit par la structuration de l'espace dans lequel sont plongées les données, comme dans le cas du *manifold learning*. Nous pensons qu'il sera utile de recourir à des relations plus complexes entré les données afin de pouvoir mieux structurer les modèles de prédication — voir une description plus détaillée dans la section 7.4.1.

Résumé — De manière plus concrète, notre projet de recherche pour les prochaines années comprend des objectifs liés à une applications particulière, comme la reconnaissance d'activités dans les vidéos, et des objectifs liés à des thématiques plus théoriques, comme l'apprentissage de modèles structurés, de modèles hiérarchiques et la minimisation de fonctions d'énergies. Ici nous

mentionnerons tout particulièrement les trois thématiques suivantes :

Modélisation et reconnaissance d'activités — le passage de la reconnaissance d'activités simples vers la reconnaissance d'activités complexes est un objectif, qui, de notre point de vue, demandera l'attention de la communauté pendant plusieurs années. Il s'agit d'un de nos thèmes de recherches principaux.

Les modèles structurés et semi-structurés — cette thématique traite la représentation d'un objet ou d'une action comme un ensemble structuré de primitives locales. Les verrous scientifiques majeurs sont i) l'intégration de relations (spatiales, temporelles, topologiques etc.) entre les primitives afin d'augmenter le pouvoir de discrimination, tout en gardant une invariance et robustesse maximale ; ii) l'apprentissage automatique robuste de ces relations à partir de données.

Segmentation, décomposition en parties, et optimisation discrète — nous comptons approfondir nos travaux sur la segmentation d'images et de maillages à l'aide d'algorithmes d'optimisation discrète, présentés dans les chapitres 5 et 6. D'un point de vue applicatif, les problèmes vont au delà des problèmes classiques associés à la segmentation, comme la restauration d'images. En effet, un lien peut être établie avec la reconnaissance d'objets et l'estimation de pose, des applications qui peuvent passer par des étapes intermédiaires de segmentation. Une autre piste concerne la régularisation d'algorithmes de segmentation sans résolution de problèmes combinatoires.

Dans les prochaines sections nous détaillerons un peu plus ces projets de recherche.

7.2 Modélisation et reconnaissance d'activités

La reconnaissance d'activités complexes et de comportements humains est une application qui nous intéresse particulièrement depuis le début du projet ANR Canada en 2008. Ce thème, qui suscite un intérêt important de la part de la communauté vision par ordinateur depuis quelques années, est riche en problèmes très intéressants d'un point de vue scientifique. Le fossé sémantique est particulièrement présent dans ce contexte, où un grand nombre d'activités peuvent être reconnues uniquement grâce au contexte. Nos recherches s'orienteront sans doute dans cette direction.

7.2.1 Modélisation d'activités dans un repère 3D

Les premiers travaux de la communauté sur la reconnaissance d'actions étaient basés sur une estimation de mouvement assez simple, par exemple sur les mouvements répétitifs. La reconnaissance d'activités de plus en plus complexes, comme par exemple les activités traitées dans le cadre de la compétition ICPR HARL 2012, organisé par le LIRIS [1], nécessite une invariance plus élevée par

1. http://liris.cnrs.fr/harl2012

rapport à l'acquisition de la vidéo. Depuis peu de temps, les caméras de profondeur ont rendu possible le traitement en 3D d'une scène. Or, les travaux existants de la communauté se focalisent sur l'usage de la profondeur pour la soustraction de fond, pour l'estimation de la pose, et pour le calcul de caractéristiques plus invariantes.

Pour les situations à l'intérieur d'un bâtiment, nous pensons qu'il sera utile de représenter une action dans un repère 3D centré sur la pièce dans laquelle elle se déroule, et non pas dans le repère de la caméra. Cela permettra une modélisation plus invariante du point de vue et du mouvement de la caméra.

7.2.2 Modélisation par parties et décompositions hiérarchiques

Comme pour la reconnaissance d'objets, les travaux sur la décomposition en parties ont eu un certain succès en reconnaissance d'actions. Il s'agit de modéliser une action par un ensemble de parties, incluant leurs positions relatives par rapport au centre de l'entité, c.à.d. de l'action, tout en gardant une souplesse suffisante dans la représentation. Pour la reconnaissance d'objets, une famille de modèles est basée sur une décomposition hiérarchique des parties apprise à partir des données d'apprentissage, généralement de manière non-supervisée, voir [FBL09] pour un exemple. L'apprentissage se fait couche par couche, les couches supplémentaires étant construites en combinant les couches précédentes. Dans ce contexte orienté image, il est intéressant de constater que les premières couches de fonctionnalités apprises par un tel système sont habituellement des filtres simples détectant des contours ou des filtres sensibles à certaines orientations et certaines échelles, tels que les filtres de Gabor. Des filtres similaires ont été trouvés dans les systèmes visuels de chat et de primates [HW59].

Nous prévoyons d'étendre ce concept aux données spatio-temporelles pour une application à la reconnaissance d'activités. Cela présente un défi, notamment à cause de la nature des données : i) dans le cas de la vidéo, le signal comprend à la fois des informations liées à l'apparence de la scène et des acteurs, et des informations liées aux mouvements. Or, ce sont les dernières qui sont intéressantes pour l'application ; ii) les parties d'une activité complexes sont liées à l'action par une position relative spatio-temporelle. Pour les actions non-triviales, la variabilité de ces positions est assez importante, rendant difficile la modélisation et l'apprentissage.

L'avantage d'une telle solution est la construction, étape par étape, d'un dictionnaire de niveau d'abstraction croissant, ce qui permet de gagner en pouvoir de discrimination par rapport à des primitives de plus bas niveau comme les points d'intérêts spatio-temporels ou le flot optique. La construction par apprentissage automatique peut éviter le travail fastidieux de construire un tel dictionnaire de manière manuelle par définition arbitraire des différents filtres.

Une extension naturelle de ce concept sera discuté dans la sous-section suivante, à savoir l'intégration de concepts de haut niveau et la modélisation de relations plus complexes.

7.2.3 Graphes sémantiques et relations topologiques

Un défi actuel majeur concerne la reconnaissance d'activités complexes, telles que les interactions entre plusieurs personnes, les interactions entre des personnes et des objets, les activités de longues durées etc. Dans ce contexte et pour ce genre d'activités, recourir à l'apprentissage automatique de la représentation complète d'une vidéo peut limiter les performances d'un système. Même les représentations hiérarchiques, décrites ci-dessus, ne seront sans doute pas capables de remplacer les

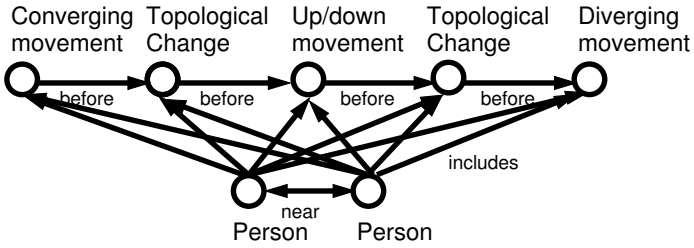


Figure 7.1 – Un modèle pour l'action « deux personnes se serrent la main » avec une modélisation très proche à la définition sémantique de l'évènement.

détecteurs de concepts de très haut niveaux actuellement utilisés. En effet, la plupart des systèmes actuels font appel un ensemble de détecteurs de haut niveau, comme les détecteurs de personnes, de visages, de certaines situations (« devant une porte ») etc. Ces détecteurs sont généralement conçus de manière spécifique pour la tâche en question, même si leur paramètres sont souvent appris à partir de données.

La question sur la manière d'extraire les caractéristiques pour la reconnaissance, soit par une conception manuelle, soit par apprentissage automatique, est sujet d'un débat depuis de nombreuses années. Les solutions par apprentissage sont bien évidemment plus commodes dans la pratique, et plus recherchées par la communauté apprentissage. Pour certaines applications, ces approches donnent des résultats très compétitifs, ce que nous avons démontré avec nos travaux sur les auto-encodeurs parcimonieux présentés dans la section 4.3.4 du chapitre 4. Cependant, tous les problèmes ne peuvent être résolus par une approche entièrement basée sur l'apprentissage automatique. Pour certains problèmes de reconnaissance, les variations de forme et d'apparence sont tellement grandes qu'il est plus efficace de concevoir un modèle adapté au problème, dont les paramètres peuvent être appris à partir de données.

Nous jugeons que l'intégration de détecteurs spécifiques reste nécessaire pour l'avenir proche. En revanche, deux problèmes restent ouverts : i) comment pouvons nous combiner ces détecteurs avec des caractéristiques de plus bas niveaux extraites de manière automatique, ce qui semble inévitable ; ii) comment pouvons nous gérer les relations spatio-temporelles et topologiques complexes entre les différents concepts, qu'ils soient sémantiques ou de bas niveaux ?

En effet, les relations spatiales et spatio-temporelles entre les parties et l'entité (objet ou action) sont habituellement modélisées par des variations probabilistes de différences de positions. En réalité, les relations entre les parties peuvent être très complexes, comme le montre l'exemple pour l'action « serrer la main » illustrée dans la figure 7.1. Cet exemple de modèle, que nous avons construit manuellement pour illustrer le problème, comprend 7 parties, dont deux correspondant aux 2 personnes impliquées dans l'action, et 5 correspondant à des mouvements spécifiques. Les relations entre parties peuvent être de nature spatiale (« X est proche de Y »), temporelle et non-linéaire ("X se passe avant Y"), topologique ("X inclut Y") etc.

Nous sommes convaincus de l'intérêt de ce genre de modélisation, même si l'exemple montré dans la figure 7.1 est sans doute naïf par rapport à une modélisation réaliste. En revanche, l'apprentissage automatique de ces modèles reste un problème assez difficile. La décomposition d'une activité en parties et en « attributs », comme sont souvent appelées les parties sémantiques d'une entité liées à un verbe, est un sujet émergent de la communauté [YJK+11, LKS11, PG11]. Par contre, les attributs sont traditionnellement gérés par approche sac de mots, c.à.d. sans mo-

délisation des relations spatio-temporelles, ou par une modélisation minimaliste, par exemple une modélisation par paires, ou par co-occurrences. La reconnaissance automatique de relations complexes entre concepts, parties et attributs reste un problème ouvert.

Ce type modélisation peut être lié à la notion du contexte, une notion bien connue en vision par ordinateur [MBC11]. Une importance particulière sera également donnée à la prise en compte du contexte de l'action : modélisation de la connaissance *a priori* de la scène, reconnaissance du type de scène etc.

7.2.4 Reconnaissance d'actions et robotique mobile

L'équipe *Imagine* du LIRIS s'est récemment doté d'une plateforme de vision intelligente autour de trois robots mobiles équipés de caméras de profondeur et d'autres capteurs. Cette plateforme, financée par l'institut INS2I du CNRS, nous sert de démonstrateur, comme banc d'essai, mais aussi comme véritable plateforme pour la conception d'algorithmes de vision mobile profitant de ce contexte particulier.

Nous comptons approfondir nos activités autour de la robotique mobile, notamment par le biais du projet *INTERABOT* porté par l'entreprise *Awabot*. La thèse de Natalia Neverova, co-encadrée par Christophe Garcia du LIRIS et financée par ce projet, et démarrant en octobre 2012, aura comme objectif la conception d'algorithmes temps réel pour la reconnaissance d'activités humaines. L'accent sera mis sur la reconnaissance d'activités complexes avec prise en compte du contexte, de la présence de certains objets dans la scène et de l'interaction de la personne avec certains objets ou avec le robot même.

Nos perspectives sur la robotique seront complétés par les travaux en perspective sur la synthèse de comportements, décrits dans les deux sous-sections suivantes.

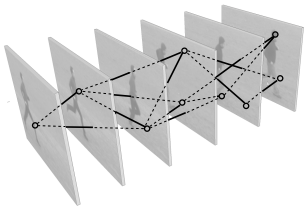
7.2.5 Self-motivation et attention

Nos travaux précédents sur la modélisation des activités humaines se sont focalisés sur la modélisation pour la reconnaissance, avec des scénarios typiquement ciblant la surveillance, les interfaces homme-machine, le chapitrage de vidéo, la création automatique de résumés de vidéos etc. Nos comptons étendre cette modélisation aux aspects génératifs, c.à.d. la modélisation pour la reconnaissance et pour la synthèse de comportements. Deux aspects nous intéressent tout particulièrement :

Attention et communication — en collaboration avec Gerard Bailly du GIPSALab, nous nous intéressons à l'apprentissage d'un modèle de comportement multimodal pour un robot humanoïde interagissant avec un partenaire humain. Il s'agit d'apprendre un comportement humain afin de pouvoir le reconnaitre et le reproduire, permettant de remplacer un partenaire humain par un robot humanöide dans le contexte d'un travail collaboratif sur une tâche commune aux deux partenaires. Le défi de ce travail est d'inscrire cette modélisation dans un cadre statistique permettant l'analyse et la génération d'activités humaines ainsi que l'adaptation en ligne des comportements appris à la variabilité des comportements du partenaire, assurant une communication fluide et le maintien de l'attention partagée entre les partenaires au cours de la tâche. La thèse d'Allaeddine Mihoub, co-encadré par les deux partenaires et démarrée en septembre 2012, est en lien avec ce sujet.

Self-motivation — en collaboration avec Olivier Georgeon de l'équipe SILEX du LIRIS, nous nous intéressons à la notion de motivation intrinsèque en robotique. Cette thématique, au LIRIS surtout développé par Olivier Georgeon, a comme objectif l'apprentissage de comportements à partir d'un système de valeurs, comme une récompense donnée après avoir atteint un objectif. Il s'agit de développer un robot mobile capable d'apprendre des comportements dans des contextes d'activités ouvertes en environnement réel. Ces activités sont menées depuis septembre 2012 dans le cadre du projet AIMOI (« Apprentissage par Interaction et MOtivation Intrinsèque dans un contexte de robotique mobile ») financé comme projet transversal par le LIRIS.

7.3 Modèles structurés et semi-structurés



7.3.1 Appariement spatio-temporel de graphes

Dans le chapitre 4, section 4.6.3, nous avons présenté une méthode d'appariement de graphes adaptée aux données spatio-temporelles. A notre connaissance nous sommes les premiers à avoir exploité les propriétés intéressantes de la dimension temporelle pour accélérer la mise en correspondance. L'algorithme présenté résout un problème d'optimisation discrète pour calculer, avec une complexité de calcul polynomiale, un optimum global du problème.

En collaboration avec Bülent Sankur de l'Université Bogazici, Istanbul, nous comptons poursuivre ces travaux en exploitant ces propriétés de manière plus approfondie. Parmi les possibilités envisagées, nous citons les trois suivantes :

Apprentissage des graphes — notre algorithme d'appariement simplifie le graphe du modèle afin de diminuer la complexité de calcul. En particulier, un point est choisi par frame du modèle, tous les points sont gardés pour chaque frame de scène. Nous prévoyons un apprentissage de ce processus de simplification afin de l'adapter à l'application. Deux pistes s'ouvrent devant nous :

i) l'intégration du choix du point comme paramètre inconnu et l'apprentissage de ce paramètre à partir des données. Un solution naturelle sera la minimisation des distance entre les graphes de la même classe et la maximisation des distances entre les graphes de classes différente. Le problème d'optimisation sous-jacent pourrait se résoudre avec un algorithme similaire à celui présenté pour l'appariement, ç.à.d. un algorithme basé sur un treillis irrégulier. A cause des variables cachées supplémentaires, liées aux nouveaux paramètres, la complexité serait toutefois plus élevée.

ii) la deuxième piste consiste à apprendre le choix des points par validation croisée en effectuant une recherche dans l'espace des graphes. Nous pensons tout particulièrement à la modélisation des solutions par modèle de mélange de graphes, un mélange correspondant à un modèle donné. Le processus itératif d'apprentissage pourrait ajouter et supprimer des graphes des différents mélanges en fonction des distances intra-classe et inter-classe.

Approches spectrales — de manière générale, la recherche d'une solution globale est préférable à la recherche d'un minimum local. Or, en pratique, même l'accélération significative obtenue par notre méthode rend assez complexe la solution exacte du problème original, nécessitant une simplification des graphes décrivant les activités humaines. Les méthodes approchées pour l'appariement de graphes calculent des minima locaux du problème, qui sont toutefois souvent d'une qualité acceptable. La possibilité de les appliquer aux problème d'origine, c.à.d. non simplifié, les rend compétitives.

Une piste que nous aimerions étudier concerne l'intégration des contraintes spatio-temporelles dans les méthodes spectrales, telles que celle de Duchenne et al. [DJP11]. En effet, la matrice d'affectation employée dans [DJP11] admet des solutions violant les contraintes que nous aimerions imposer, indiquant une possibilité de diminuer l'espace de recherche. Nos études préliminaires ont montré, que la matrice se laisse décomposer en plusieurs matrices de petites taille pour lesquelles toutes les réalisations sont admissibles. En théorie, un algorithme spectrale basé sur cette décomposition devrait permettre d'obtenir des meilleurs solutions avec un complexité de calcul inférieure.

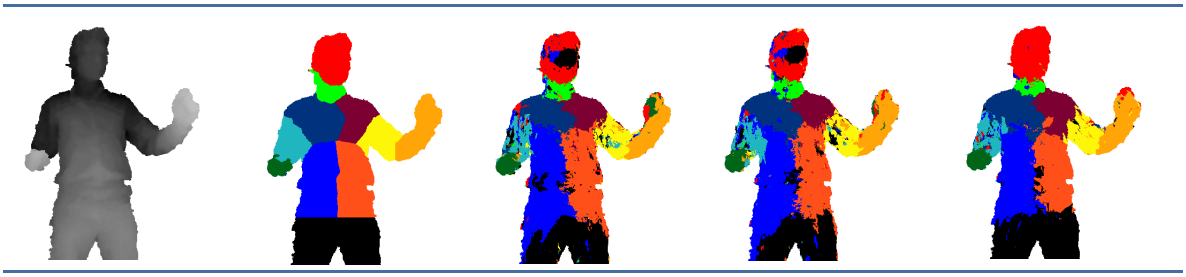
Isomorphisme de sous-graphes — une piste complémentaire est basée sur un principe diamétralement opposé à celui décrit ci-dessus : au lieu de rendre le système plus souple en calculant une solution approchée, nous proposons de calculer la solution exacte pour le problème exact, à savoir un isomorphisme de sous-graphes. Rappelons que dans la méthode proposée dans la section 4.6.3 du chapitre 4, nous avons calculé la solution exacte pour un appariement approché, c.à.d. un appariement qui ne respecte pas nécessairement un isomorphisme entre les graphes.

La restriction aux appariements respectant un sous-isomorphisme diminuera de manière significative l'espace de recherche. La différence avec les méthodes existantes de calcul d'isomorophismes de sous-graphes réside dans l'intégration des contraintes spatio-temporelles. Nous pensons que la complexité moyenne pour les graphes typiquement rencontrés dans le cadre applicatif sera bien inférieure à la complexité de calcul dans le pire des cas, rendant cette approche compétitive. Une réponse plus concluante nécessitera des études et des expériences approfondies, que nous comptons effectuer en collaboration avec Christine Solnon du LIRIS.

7.4 Segmentation, décomposition en parties, et optimisation discrète

7.4.1 Régularisation spatiale sans termes par paires

Nous pensons que la recherche apportera des solutions au problème de segmentation qui se passeront d'une modélisation par contraintes de deuxième ordre. L'exemple de la segmentation d'images en profondeur dans le cadre du projet MS Kinect [SFC+11] a montré, qu'une segmentation basée



uniquement sur des classifieurs donnant des résultats indépendants par pixel peut suffire pour résoudre un problème très difficile. En revanche, cela nécessite un support assez grand sur lequel sont calculées les caractéristiques, et une base d'apprentissage de très grande taille — $2.000.000.000$ vecteurs ont été utilisés pour le projet Kinect, nécessitant l'exécution de l'algorithme d'apprentissage sur un cluster de calcul comprenant 1000 processeurs.

Bien sur cela ne sera pas la solution miracle à tous nos problèmes. Pour un grand nombre d'applications il est trop difficile ou impossible de collectionner une quantité suffisante de données d'apprentissage pour que cette approche naïve puisse donner des résultats corrects. Par contre, nous sommes convaincus que les approches basées sur les classifieurs peuvent être améliorées en structurant le modèle de prédiction. En effet, les informations *a priori* modélisées traditionnellement par des modèles graphiques peuvent aussi servir à restreindre les modèles de prédiction comme ceux des classifieurs de type MLP, SVM, forêt aléatoire etc. Le raisonnement est similaire à celui du *manifold learning* : malgré le fait que les caractéristiques d'un problème sont embarquées dans un espace vectoriel de haute dimension, les structures du problème se trouvent sur une variété de dimension généralement beaucoup plus petite.

Nous avons présenté nos premiers travaux sur ce sujet dans le chapitre 5, section 5.9, où nous avons enrichi le modèle de prédiction d'un classifieur de type forêt aléatoire en injectant les informations sur les relations spatiales entre les parties d'un objet. Nous avons appelé ce mode d'apprentissage « apprentissage spatial ». Nous comptons approfondir ces travaux dans l'avenir ; deux pistes sont envisagées pour le moment :

Apprentissage spatiale par réseaux de neurones convolutionnels — en collaboration avec Graham Taylor de l'Université de Guelph, Canada, et dans le cadre de la thèse de Mingyuan Jiu, nous sommes actuellement en train d'étendre nos travaux sur les forêts aléatoires aux réseaux de neurones convolutionnels. Au delà d'un simple changement de classifieur, ces travaux ont comme objectif d'injecter les relations spatiales de deux manières différentes, propres aux modèles profonds : l'apprentissage non-supervisé, couche par couche ; et l'apprentissage supervisé, en rétro-propageant le gradient de l'erreur.

Régularisation spatiale par modèles causaux hiérarchiques — une piste plus long terme est basée sur un compromis entre les deux extrêmes, à savoir, d'une part une modélisation à l'aide de termes par paires, par exemple par un MRF ; et d'autre part, une modélisation à l'aide de termes unaires uniquement, comme par exemple dans nos travaux sur les forêts aléatoires. Nous envisageons de travailler sur un modèle causal intégrant des termes par paires de manière efficace ne nécessitant pas la résolution d'un problème d'optimisation discrète lors de l'inférence.

L'idée est la suivante : un classifieur de type forêt aléatoire est appris pour classifier chaque

pixel d'une image. Lors de l'inférence, le système complet pour toute une image est résolu par un parcours en largeur. De cette manière, la décision pour un pixel donné et pour une couche donnée (de choisir le sous-arbre gauche ou droite) peut potentiellement se servir des décisions de tous les autres pixels jusqu'au niveau précédent. La manière d'exploiter cette information (connectivité et paramètres) pourrait être apprise à partir des données lors de la phase d'apprentissage.

7.4.2 Segmentation en préservant les frontières

Dans ce mémoire nous avons décrit, entre autres, nos travaux sur la segmentation d'images, de vidéos et de maillages surfaciques en région homogènes selon un certain critère. Ci-dessus nous avons mentionné des approches basées sur une classification des pixels sans contraintes de deuxième ordre, c.à.d. sans terme de régularisation nécessitant la résolution d'un problème d'optimisation discrète. De par leur vitesse, ces méthodes sont adaptées aux applications interactives, comme par exemple l'estimation de la pose humaine pour une interface homme-machine. En revanche, si un calcul interactive n'est pas demandé, il est possible de favoriser la qualité du résultat en ajoutant des termes de régularisation, par exemple par le framework d'un MRF. Nous avons proposé des telles solutions pour les images et les vidéos (voir le chapitre 5) et pour les maillages (voir le chapitre 6).

Afin d'éviter un lissage trop important du résultat, il peut être intéressant de modéliser, de manière explicite, les frontières entre les régions dans l'algorithme d'optimisation discrète, comme nous l'avons proposé pour la décomposition de maillages dans la section 6.2. Cela permet de désactiver la régularisation d'une paire de régions voisines si elles sont séparées par une frontière. Or, ce genre de modèle produit des fonctions d'énergie non-sous-modulaires difficiles à minimiser.

Nous jugeons que la modélisation par approches hiérarchiques est une piste très prometteuse dans ce contexte. L'intégration rapide d'indications venant du bas niveau (pixels d'une image ou sommets d'un maillage) vers des niveaux intermédiaires et élevés permet une régularisation plus efficace. En revanche, les modèles hiérarchiques traditionnels ne permettent pas de prendre en comptes les frontières, comme par exemple l'arbre quaternaire [LPH00] ou notre Markov cube, présenté dans le chapitre 5. La modélisation hiérarchique conjointe d'un processus aléatoire sur une grille et d'un processus sur les frontières sera donc un de nos sujets de recherches dans l'avenir.

Bibliographie

[AAGU07] P. Alliez, M. Attene, C. Gotsman, and G. Ucelli. *Recent Advances in Remeshing of Surfaces*, chapter Shape Analysis and Structuring. Springer Verlag, 2007.

[AdVDI05] P. Alliez, E. Colin de Verdière, O. Devilliers, and M. Isenburg. Centroidal voronoi diagrams for isotropic surface remeshing. *Graphical Models*, 67(3) :204–231, 2005.

[AFS06] M. Attene, B. Falcidieno, and M. Spagnuolo. Hierarchical mesh segmentation based on fitting primitives. *Visual Computer*, 22 :181–193, 2006.

[AFSW03] M. Attene, B. Falcidieno, M. Spagnuolo, and G. Wyvill. A mapping-independent primitive for the triangulation of parametric surfaces. *Graphical models*, 65(5) :260–273, September 2003.

[AHL65] K. Abend, T.J. Harley, and L.N.Kanal. Classification of binary random patterns. *IEEE Transactions on Information Theory*, IT-11(4) :538–544, 1965.

[AKM+06] M. Attene, S. Katz, M. Mortara, G. Patane', M. Spagnuolo, and A. Tal. Mesh segmentation — a comparative study. In *International Conference on Shape Modeling and Applications*, 2006.

[AMD02] P. Alliez, M. Meyer, and M. Desbrun. Interactive geometry remeshing. *ACM Transactions on Graphics*, 21(3) :347–354, 2002.

[AS07] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Transactions on Graphics*, 26(3) :10, 2007.

[BAT12] F. Bergamasco, A. Albarelli, and A. Torsello. A graph-based technique for semi-supervised segmentation of 3d surfaces. *Pattern Recognition Letters*, 33(15) :2057 – 2064, 2012.

[BB12] P. Bilinski and F. Brémond. Statistics of pairwise cooccurring local spatiotemporal features for human action recognition. In *International ECCV Workshop on Video Event Categorization, Tagging and Retrieval*, 2012.

[BBBS09] L. Ballan, M. Bertini, A. Del Bimbo, and G. Serra. Action categorization in soccer videos using string kernels. In *International Conference on Content Based Multimedia Indexing (CBMI)*, 2009.

[Bel94] M.G. Bello. A combined Markov random field and wave-packet transform-based approach for image segmentation. *IEEE transactions on image processing*, 3(6) :834–846, 1994.

[Bes74] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, 36(2) :192–236, 1974.

[Bes75] J. Besag. Statistical analysis of non-lattice data. *The Statistician*, 24(3) :179–195, 1975.

[Bis06] C.M. Bishop. *Pattern Recognition and Machine learning*. Springer Verlag, 2006.

[BJ99] S. Bres and J.M. Jolion. Detection of interest points for image indexing. In *3rd Int. Conf. on Visual Inf. Systems, Visual 99*, pages 427–434. Springer, Lecture Notes in Computer Science, 1614, June 1999.

[BK04] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max- flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9) :1124–1137, 2004.

[BLVD11] H. Benhabiles, G. Lavoué, J.-P. Vandeborre, and M. Daoudi. Learning Boundary Edges for 3D-Mesh Segmentation. *Computer Graphics Forum*, June 2011.

[BM97] H. Bunke and B.T. Messmer. Recent advances in graph matching. *Journal of Pattern Recognition and Artificial Intelligence*, 11(1) :169–203, 1997.

[BMW+10a] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Action classification in soccer videos with long short-term memory recurrent neural networks. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 154–159, 2010.

[BMW+10b] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Une approche neuronale pour la classification d'actions de sport par la prise en compte du contenu visuel et du mouvement dominant. In *COmpression et REprésentation des Signaux Audiovisuels (CORESA)*, 2010.

[BMW+11] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding : Inducing Behavioral Change*, 2011.

[BMW+12a] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sparse shift-invariant representation of local 2d patterns and sequence learning for human action recognition. In IEEE, editor, *International Conference on Pattern Recognition (ICPR)*, 2012.

[BMW+12b] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Spatio-temporal convolutional sparse auto-encoder for sequence classification. In *British Machine Vision Conference (BMVC)*, 2012.

[BOP97] M. Brand, N. Oliver, and A. Pentland. Coupled hidden markov models for complex action recognition. In IEEE, editor, *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 994–998, 1997.

[BP93] B. Braathen and W. Pieczynski. Global and Local Methods of Unsupervised Bayesian Segmentaiton of Images. *Machine Graphics and Vision*, 2(1) :39–52, 1993.

[BS94] C.A. Bouman and M. Shapiro. A Multiscale Random Field Model for Bayesian Image Segmentation. *IEEE Transactions on Image Processing*, 3(2) :162–177, 3 1994.

[BT11] W. Brendel and S. Todorovic. Learning spatiotemporal graphs of human activities. In *ICPR*, 2011.

[BVZ01] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11) :1222–1239, 2001.

[CBCFfY11] T. Crivelli, P. Bouthemy, B. Cernuschi-Frias, and J. f. Yao. Simultaneous motion detection and background reconstruction with a conditional mixed-state markov random field. *International Journal on Computer Vision*, 94 :295–316, 2011.

[CCO+10] A. Carlier, V. Charvillat, W.T. Ooi, R. Grigoras, and G. Morin. Crowd-sourced automatic zoom and scroll for video retargeting. In *ACM Multimedia*, 2010.

[CFL11] C. Chen, D. Freedman, and C. H. Lampert. Enforcing topological constraints in random field image segmentation. In *International Conference on Computer Vision and Pattern Recognition*, 2011.

[CGPP03] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25(10) :1337–1342, 2003.

[CSAD04] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *Proceedings of ACM SIGGRAPH*, pages 905 – 914, 2004.

[CSM03] D. Cohen-Steiner and J. Morvan. Restricted delaunay triangulations and normal cycle. In *19th Annu. ACM Sympos. Comput. Geom.*, 2003.

[CWS12a] O. Celiktutan, C. Wolf, and B. Sankur. Real-time exact graph matching with application in human action recognition. In *International Workshop on Human Behavior Understanding*, 2012.

[CWS12b] O. Celiktutan, C. Wolf, and B. Sankur. Fast exact matching and correspondence with hyper-graphs on spatio-temporal data. Technical Report RR-LIRIS-2012-002, Laboratoire d'informatique en images et systèmes d'information, February 2012.

[DAGG11] A. Dhall, A. Asthana, R. Goecke, and T. Gedeon. Emotion recognition using phog and lpq features. In *International Conference on Automatic Face & Gesture Recognition*, pages 878–883, 2011.

[DBKP09] O. Duchenne, F.R. Bach, I.-S. Kweon, and J. Ponce. A tensor-based algorithm for high-order graph matching. In *CVPR*, pages 1980–1987, 2009.

[DE87] H. Derin and H. Elliott. Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1) :39–55, 1987.

[DEK85] H. Derin, H. Elliott, and J. Kuang. A New Approach to Parameter Estimation for Gibbs Random Fields. In *Proceedings of the International Conference on Accoustics, Speech and Signal Processing*, volume 10, pages 913–916, 1985.

[DJP11] O. Duchenne, A. Joulin, and J. Ponce. A graph-matching kernel for object categorization. In *ICCV*, 2011.

[DKG05] S. Dong, S. Kircher, and M. Garland. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Computer Aided Geometric Design*, 22(5) :392–423, 2005.

[DL11] F. Drira and F. Lebourgeois. A new PDE-based approach for singularity-preserving regularization : application to degraded characters restoration . *International Journal on Document Analysis and Recognition (IJDAR)*, 2011.

[DLE06a] F. Drira, F. LeBourgeois, and H. Emptoz. A modified mean shift algorithm for efficient document image restoration. In *Proceedings of the IEEE/ACM International Conference on Signal Image Technology and Internet Based Systems*, pages 686–695, 2006.

[DLE06b] F. Drira, F. LeBourgeois, and H. Emptoz. Restoring ink bleed-through degraded document images using a recursive unsupervised classification technique. In *Proceedings of the 7th Workshop on Document Analysis Systems*, pages 38–49, 2006.

[DLE07] F. Drira, F. Lebourgeois, and H. Emptoz. OCR accuracy improvement through a PDE-based approach. In *International Conference on Document Analysis and Recognition*, volume 2, pages 1068–1072, 2007.

[dlHC00] C. de la Higuera and F. Casacuberta. Topology of strings : median string is np-complete. *Theoretical Computer Science*, 230(1-2) :39–48, 2000.

[DLR77] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1) :1–38, 1977.

[DLR05] T.K. Dey, G. Li, and T. Ray. Polygonal surface remeshing with Delaunay refinement. volume 26, pages 343–361. Springer Verlag, 2005.

[DM05] K. Donaldson and G.K. Myers. Bayesian super-resolution of text in video with a text-specific bimodal prior. *International Journal on Document Analysis and Recognition*, 7(2-3) :159–167, 2005.

[DMSB99] M. Desbrun, M. Meyer, P. Schröder, and A.H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 317–324, 1999.

[DOIB10] A. Delong, A. Osokin, H.N. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2173–2180, 2010.

[Don00] H.-S. Don. A noise attribute thresholding method for document image binarization. *International Journal on Document Analysis and Recognition*, 4(2) :131–138, 2000.

[DP01] E. Dubois and A. Pathak. Reduction of bleed-through in scanned manuscript documents. In *Proceedings of the Image Processing, Image Quality, Image Capture Systems Conference*, pages 177–180, 2001.

[DPFH10] A. Delaunoy, E. Prados, K. Fundana, and A. Heyden. Segmentation convexe multi-région de données sur les surfaces. In *17ème Congrès de Reconnaissance des Formes et Intelligence Artificielle*, 2010.

[DPS89] D.M.Greig, B.T. Porteous, and A.H. Seheult. Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society B*, 51(2) :271–279, 1989.

[DRCB05] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *ICCV VS-PETS*, Beijing, China, 2005.

[DS06] J. Darbon and M. Sigelle. Image restoration with discrete constrained total variation part ii : Levelable functions, convex priors and non-convex cases. *Journal of Mathematical Imaging and Vision*, 26(3) :277–291, 2006.

[FB81] M.A. Fischler and R.C. Bolles. Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24 :381–395, 1981.

[FBL09] S. Fidler, M. Boben, and A. Leonardis. *Object Categorization : Computer and Human Vision Perspectives*, chapter Learning Hierarchical Compositional Representations of Object Structure. Cambridge University Press, 2009.

[FCNL12] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Scene parsing with multiscale feature learning, purity trees, and optimal covers. In *International Conference on Machine Learning (ICML)*, 2012.

[FDP+03] R. Fjortoft, Y. Delignon, W. Pieczynski, M. Sigelle, and F. Tupin. Unsupervised classification of radar images using hidden Markov chains and hidden Markov random fields. *IEEE Transaction on Geoscience and remote sensing*, 41(3) :675–686, 2003.

[FE73] M.A. Fischler and R.A. Elschlager. The representation and matching of pictorial structures. *IEEE Transactions on Computer*, 22(1) :67–92, 1973.

[FF62] L.R. Ford and D.R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.

[FGMR10] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(9) :1627–1645, 2010.

[FH05] P.F. Felzenszwalb and D.P. Huttenlocher. Pictorial Structures for Object Recognition. *International Journal of Computer Vision*, 61(1) :55–79, 2005.

[FMJZ08] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Progressive search space reduction for human pose estimation. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[FSV01] P. Foggia, C. Sansone, and M. Vento. A performance comparison of five algorithms for graph isomorphism. In *IAPR TC-15 Workshop on Graph-based Representations in Pattern Recognition*, page 188–199, 2001.

[FTG06] V. Ferrari, T. Tuytelaars, and Luc J. Van Gool. Object detection by contour segment networks. In *ECCV (3)*, pages 14–28, 2006.

[GBS+07] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(12) :2247–2253, 2007.

[GCHA10] Z. Gao, M.Y. Chen, A. Hauptmann, and A.Cai. Comparing evaluation protocols on the kth dataset. In *Human Behavior Understanding*, volume LNCS 6219, pages 88–100, 2010.

[GCO06] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Trans. Graph.*, 25(1) :130–150, 2006.

[GD05] K. Grauman and T. Darrell. Pyramid match kernels : Discriminative classification with sets of image features. In *International Conference on Computer Vision (ICCV)*, 2005.

[GG84] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6) :721–741, 11 1984.

[GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness*. W.H. Freeman, 1979.

[GLS+09] Y. Guo, F. Liu, J. Shi, Z.-H. Zhou, and M. Gleicher. Image retargeting using mesh parametrization. *IEEE Transactions on Multimedia*, 11(5) :856–867, 2009.

[GNP11] B. Gatos, K. Ntirogiannis, and I. Pratikakis. Dibco 2009 : document image binarization contest. *International Journal on Document Image Analysis and Recognition*, 14 :35–44, 2011.

[Gui07] G. Guillaume. A roughness measure for 3d mesh visual masking. In *ACM Symposium on Applied Perception in Graphics and Visualization*, pages 57–60, 2007.

[GWH01] M. Garland, A. Willmott, and P. Heckbert. Hierarchical face clustering on polygonal surfaces. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 49 – 58, 2001.

[GZSRC11] U. Gaur, Y. Zhu, B. Song, and A. Roy-Chowdhury. A "string of feature graphs" model for recognition of complex activities in natural videos. In *International Conference on Computer Vision*, 2011.

[HC68] J.M. Hammersley and P. Clifford. Markov fields on finite graphs and lattices. unpublished manuscript, 1968.

[HDD+93] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *SIGGRAPH*, pages 19–26, 1993.

[HHS84] P.L. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Mathematical Programming*, 28 :121–155, 1984.

[HKG11] Q. Huang, V. Koltun, and L. Guibas. Joint shape seg mentation with linear programming. *ACM Transactions on Graphics*, 30(6), 2011.

[Hop97] H. Hoppe. View-dependent refinement of progressive meshes. In *SIGGRAPH*, pages 189–198, 1997.

[HRW07] D. Hoiem, C. Rother, and J. Winn. 3D layoutCRF for multi-view object class recognition and segmentation. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, 2007.

[HS97] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8) :1735–1780, 1997.

[HW59] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *Journal of Physiology*, 148 :574–591, 1959.

[HW12] E.R. Hancock and R.C. Wilson. Pattern analysis with graphs : Parallel work at bern and york. *Pattern Recognition Letters*, 33(2012) :833–841, 2012.

[IB11] H. Isack and Y. Boykov. Energy-based Geometric Multi-Model Fitting. *International Journal of Computer Vision*, pages 1–25, 2011.

[Ish03] H. Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10) :1333–1336, 2003.

[JWB13] M. Jiu, C. Wolf, and A. Baskurt. Integrating spatial layout of object parts into classification without pairwise terms : application to fast body parts estimation from depth images. In *International Conference on Computer Vision Theory and Applications (VISAPP)*, 2013.

[JWGB12] M. Jiu, C. Wolf, C. Garcia, and A. Baskurt. Supervised learning and codebook optimization of bag of words models. *Cognitive Computation*, 4 :409–419, 2012.

[KA94] S.-S. Kuo and O.E. Agazzi. Keyword spotting in poorly printed documents using pseudo 2-d hidden Markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8) :842–848, 1994.

[KB12] M. Kaaniche and F. Brémond. Recognizing gestures by learning local motion signatures of hog descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11) :2247–2258, 2012.

[KBZ96] Z. Kato, M. Berthod, and J. Zerubia. A hierarchical Markov random field model and multitemperature annealing for parallel image classification. *Graphical Models and Image Processing*, 58(1) :18–37, 1996.

[KFL01] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE transactions on Information Theory*, 47(2) :498–519, 2001.

[KH05] S. Kumar and M. Hebert. A hierarchical field framework for unified context-based classification. In *International Conference on Computer Vision*, volume 2, pages 1284– 1291, 2005.

[KH10] E. Kalogerakis and A. Hertzman. Learning 3d mesh segmentation and labeling. *ACM Transactions on Graphics*, pages 1–12, 2010.

[KLT05] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21(8-10) :649–658, 2005.

[KR07] V. Kolmogorov and C. Rother. Minimizing non-submodular functions with graph cuts - a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(7), 2007.

[KT03] S. Katz and A. Tal. Hierarchical mesh decomposition using fuzzy clustering and cuts. In *ACM SIGGRAPH*, pages 954 – 961, 2003.

[KZ04] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts ? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2) :147–159, 2004.

[LAM+11] B. Li, M. Ayazoğlu, T. Mao, O. I. Camps, and M. Sznaier. Activity recognition using dynamic subspace angles. In *CVPR*, 2011.

[LCL11] J. Lee, M. Cho, and K.M. Lee. Hyper-graph matching via reweighted random walks. In *CVPR X*, 2011.

[LDB05] G. Lavoué, F. Dupont, and A. Baskurt. A new cad mesh segmentation method, based on curvature tensor analysis. *Computer-Aided Design*, 37(10) :975–987, 2005.

[Lev66] A. Levenstein. Binary codes capable of correcting deletions, insertions and reversals. *Sov. Phy. Dohl.*, 10 :707–710, 1966.

[LH05] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *International conference on computer Vision (ICCV)*, 2005.

[Li01] S.Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer Verlag, 2001.

[LJW+10] P.Y. Laffont, J.Y. Jun, C. Wolf, Y.W. Tai, K. Idrissi, G. Drettakis, and S.-E. Yoon. Interactive content-aware zooming. In *Graphics Interface*, 2010.

[LKS11] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3337–3344, 2011.

[LL03] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, pages 432–439, 2003.

[LLE04] Y. Leydier, F. LeBourgeois, and H. Emptoz. Serialized Unsupervised Classifier for Adaptative Color Image Segmentation : Application to Digitized Ancient Manuscripts. In *International Conference on Pattern Recogniton*, pages 494–497, 2004.

[LLF05] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[LLLP08] P. Lanchantin, J. Lapuyade-Lahorgue, and W. Pieczynski. Unsupervised segmentation of triplet markov chains hidden with long-memory noise. *Signal Processing*, 88(5) :1134–1151, 2008.

[LMP01] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields : Probabilistic models for segmenting and labeling data. In *International Conference on Machine Learning*, 2001.

[LMSR08] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, pages 1–8, 2008.

[Low99] D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, 1999.

[LP92] E. Levin and R. Pieraccini. Dynamic planar warping for optical character recognition. In *Proceedings of the International Conference on Accoustics, Speech and Signal Processing*, volume 3, pages 149–152, 1992.

[LPH00] J.-M. Laferte, P. Perez, and F. Heitz. Discrete Markov image modelling and inference on the quad tree. *IEEE Transactions on Image Processing*, 9(3) :390–404, 2000.

[LPRM02] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph.*, 21(3) :362–371, 2002.

[LPS+05] S.M. Lucas, A. Panaretos, L. Sosa, A. Tang, S. Wong, R. Young, K. Ashida, H. Nagai, M. Okamoto, H. Yamamoto, H. Miyao, J. Zhu, W. Ou, C. Wolf, J.-M. Jolion, L. Todoran, M. Worring, and X. Lin. ICDAR 2003 Robust Reading Competitions : Entries, Results and Future Directions. *International Journal on Document Analysis and Recognition - Special Issue on Camera-based Text and Document Recognition*, 7(2-3) :105–122, 2005.

[LRR08] V. Lempitsky, S. Roth, and C. Rother. Fusionflow : Discrete-continuous optimization for optical flow estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2008.

[LRRB10] V. Lempitsky, C. Rother, C. Roth, and A. Blake. Fusion moves for markov random field optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8) :1392–1405, 2010.

[LS88] S.L. Lauritzen and D.J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society B*, 50 :157–224, 1988.

[LS08] J. Liu and M. Shah. Learning human actions via information maximization. In *CVPR*, Los Alamitos, CA, 2008.

[LSP06] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features : Spatial pyramid matching for recognizing natural scene categories. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.

[LTJW07] L. Liu, C.-L. Tai, Z. Ji, and G. Wang. Non-iterative approach for global mesh optimization. *Computer-Aided Design*, 39(9) :772–782, 2007.

[LVJ05] C. Lee, A. Varshney, and D. Jacobs. Mesh saliency. In *ACM Siggraph*, pages 659–666, 2005.

[LW08] G. Lavoué and C. Wolf. Markov random fields for improving 3d mesh analysis and segmentation. In *Proceedings of the EUROGRAPHICS 2008 Workshop on 3D Object Retrieval*, 2008.

[LWH03] B. Luo, C. Wilson, and E.R. Hancok. Spectral embedding of graphs. *Pattern Recognition*, 36(10) :2213–2230, 2003.

[LWW+11] G. Littlewort, J. Whitehill, T.F. Wu, N. Butko, P. Ruvolo, J. Movellan, and M. Bartlett. The motion in emotion — a cert based approach to the fera emotion challenge. In *International Conference on Automatic Face & Gesture Recognition*, pages 897–902, 2011.

[LZ04] R. Liu and H. Zhang. Segmentation of 3d meshes through spectral clustering. In *Pacific Conference on Computer Graphics and Applications*, pages 298–305, 2004.

[LZH+07] Y.-K. Lai, Q.-Y. Zhou, S.-M. Hu, J. Wallner, and H. Pottmann. Robust feature classification and editing. *IEEE Transactions on Visualization and Computer Graphics*, 13(1) :34–45, 2007.

[LZLZ09] L. Lin, K. Zeng, X. Liu, and S.-C. Zhu. Layered graph matching by composite cluster sampling with collaborative and competitive interactions. *CVPR*, 0 :1351–1358, 2009.

[LZS11] M. Leordeanu, A. Zanfir, and C. Sminchisescu. Semi-supervised learning and optimization for hypergraph matching. In *ICCV 2011*, 2011.

[MB99] B.T. Messmer and H. Bunke. A decision tree approach to graph and subgraph isomorphism detection. *Pattern Recognition*, 32(12) :1979–1998, 1999.

[MBC11] O. Marques, E. Barenholtz, and V. Charvillat. Context modeling in computer vision : techniques, implications and applications. *Multimedia Tools and Applications*, 2011.

[McK81] B.D. McKay. Practical graph isomorphism. In *Conference on Numerical Mathematics and Computing : Congressus Numerantium*, volume 30, pages 45–87, 1981.

[MCPB00] M. Mignotte, C. Collet, P. Perez, and P. Bouthemy. Sonar image segmentation using an unsupervised hierarchical mrf model. *IEEE Transactions on Image Processing*, 9(7) :1216–1231, 2000.

[MRPBB11] H. Meng, B. Romera-Paredes, and N. Bianchi-Berthouze. Emotion recognition by two view svm 2k classifier on dynamic facial expression features. In *International Conference on Automatic Face & Gesture Recognition*, pages 854–859, 2011.

[MS85] D. Mumford and J Shah. Boundary detection by minimizing functionals. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22–26, 1985.

[MS93] D. Milun and D. Sher. Improving Sampled Probabilty Distributions for Markov Random Fields. *Pattern Recognition Letters*, 14(10) :781–788, 1993.

[MSW09] M. Mouret, C. Solnon, and C. Wolf. Classification of images based on hidden markov models. In *International workshop on content-based multimedia indexing*, 2009.

[MU11] K. Mikolajczyk and H. Uemura. Action recognition with appearance–motion features and fast search trees. *Computer Vision and Image Understanding*, 115(3) :426–438, March 2011.

[MWJ99] K. Murphy, Y. Weiss, and M. I. Jordan. Loopy belief-propagation for approximate inference : An empirical study. In *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 467–475, 1999.

[NCFF10] J. C. Niebles, C. W. Chen, and L. Fei-Fei. Modelling temporal sturcture of decomposable motion segments for activity classification. In *ECCV*, pages 1–14, 2010.

[NFF07] J. C. Niebles and L. Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *CVPR*, pages 1–8, 2007.

[NHH07] H. Ning, Y. Hu, and T. S. Huang. Searching human behaviours using spatial-temporal words. In *ICIP*, volume 6, pages 337–340, 2007.

[Nib86] W. Niblack. *An Introduction to Digital Image Processing*, pages 115–116. Prentice Hall, 1986.

[NJ02] A.Y. Ng and M.I. Jordan. On discriminative vs. generative classifiers : A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing (NIPS)*, page 841–848, 2002.

[NS02] H. Nishida and T. Suzuki. Correcting show-through effects on document images by multiscale analysis. In *Proceedings of the International Conference on Pattern Recognition*, volume 3, pages 65–68, 2002.

[Par88] G. Parisi. *Statistical Field Theory*. Addison-Wesely, 1988.

[PB03] P.P. Pébay and T.J. Baker. Analysis of triangle quality measures. *Mathematics of Computation*, 72(244) :1817–1840, 2003.

[PCR+04] J.-N. Provost, C. Collet, P. Rostaing, P. Pérez, and P. Bouthemy. Hierarchical markovian segmentation of multispectral images for the reconstruction of water depth maps. *Computer Vision and Image Understanding*, 93(2) :155–174, 2004.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufman, San Mateo, 1988.

[PF08] D.H. Parks and S.S. Fels. Evaluation of background subtraction algorithms with post-processing. In *Proceedings of the International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 192–199, 2008.

[PFNK94] P. Pudil, F. J. Ferri, J. Novovicov, and J. Kittler. Floating search methods for feature selection with non-monotonic criterion functions. In *ICPR*, pages 279–283, 1994.

[PG11] D. Parikh and K. Grauman. Relative attributes. In *International Conference on Computer Vision (ICCV)*, 2011. Marr Prize (Best Paper Award) Winner.

[Pie03] W. Pieczynski. Pairwise Markov chains. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(5) :634–639, 2003.

[Pie07] W. Pieczynski. Convergence of the iterative conditional estimation and application to mixture proportion identification. In *IEEE/SP Workshop on Statistical Signal Processing*, pages 49–53, 2007.

[Pot52] R.B. Potts. Some generalized order-disorder transformations. *Mathematical Proceedings*, 48(1) :106–109, 1952.

[PR12] H. Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[PS06] P.W. Power and J. A. Schoonees. Understanding background mixture models for foreground segmentation. *Image and Vision Computing*, 24(5), 2006.

[PT00] W. Pieczynski and A.-N. Tebbache. Pairwise Markov random fields and segmentation of textured images. *Machine Graphics & Vision*, 9(3) :705–718, 2000.

[PW04] S.E. Pav and N.J. Walkington. Robust three dimensional Delaunay refinement. In *Thirteenth International Meshing Roundtable*, pages 145–156, 2004.

[QCD05] A. Quattoni, M. Collins, and T. Darrell. Conditional random fields for object recognition. In Lawrence K. Saul, Yair Weiss, and Leon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1097–1104. MIT Press, 2005.

[QWM+07] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, and T. Darrell. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis an Machine Intelligence (PAMI)*, 29(10) :1848–1852, 2007.

[RA09] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match : video structure comparison for recognition of complex human activities. In *ICCV*, 2009.

[Rab89] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–286, 1989.

[RBB+12] G. Rosman, M. Bronstein, A. Bronstein, A. Wolf, and R. Kimmel. Group-valued regularization framework for motion segmentation of dynamic non-rigid shapes. In *Proceedings of the Third international conference on Scale Space and Variational Methods in Computer Vision*, pages 725–736, 2012.

[RBBT11] R. Romdhane, B. Boulay, F. Bremond, and M. Thonnat. Probabilistic recognition of complex event. In *International Conference on Computer Vision Systems*, 2011.

[Reg95] J.C. Regin. *Développement d'outils algorithmiques pour l'intelligence artificielle Texte imprimé : Application à la chimie organique / par Jean-Charles Régin*. PhD thesis, Université de Montpellier 2, 1995.

[RHBL07] M.A. Ranzato, F.J. Huang, Y.L. Boureau, and Y. Lecun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[RKLS07] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *International Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[RL+07] J. Revaud, G. Lavoué, , Y. Ariki, and A. Baskurt. Fast and cheap object recognition by linear combination of views. *International Conference on Image and Video Retrieval*, 2007.

[RLB09] J. Revaud, G. Lavoué, and A. Baskurt. Improving zernike moments comparison for optimal similarity and rotation angle retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4) :627–636, 2009.

[RLJS05] J. Ros, C. Laurent, J.-M. Jolion, and I. Simand. Comparing string representations and distances in a natural images classification task. In *GbRPR*, volume 3434 of *Lecture Notes in Computer Science*, pages 72–81. Springer, 2005.

[ROF92] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60 :259–268, 1992.

[RSA08] M. Rubinstein, A. Shamir, and S. Avidan. Improved seam carving for video retargeting. In *SIGGRAPH*, pages 1–9, 2008.

[RTG00] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2) :99–121, 2000.

[SAG03] V. Surazhsky, P. Alliez, and C. Gotsman. Isotropic remeshing of surfaces : a local parameterization approach. In *Proceedings of the 12th International Meshing Roundtable*, number Sandia National Laboratories, pages 215–224, 2003.

[SAS07] P. Scovanner, S. Ali, and M. Shah. A 3d sift descriptor and its application to action recognition. In *ACM Multimedia*, pages 357–360, New York,USA, 2007.

[SAY75] G. Salton, A.Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of ACM*, 18(11) :613–620, 1975.

[SB91] M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1) :11–32, 1991.

[SBC05] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *ICCV*, pages 503–510, 2005.

[SBCBG11] J. Solomon, M. Ben-Chen, A. Butscher, and L. Guibas. As-killing-as-possible vector fields for planar deformation. *Computer Graphics Forum (Eurographics)*, 30(5) :1543–1552, 2011.

[SC06] F. Salzenstein and C. Collet. Fuzzy Markov random fields versus chains for multispectral image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11), 2006.

[SDNFF08] S. Savarese, A. Delpozo, J. Niebles, and L. Fei-Fei. Spatial-temporal correlatons for unsupervised action classification. In *WMVC*, Los Alamitos, CA, 2008.

[SFC+11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *International Conference on Pattern Recognition and Computer Vision*, 2011.

[SG00] C. Stauffer and W.E.L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(8) :747–757, 2000.

[SG03] V. Surazhsky and C. Gotsman. Explicit surface remeshing. In *Proceedings of the Eurographics Symposium on Geometry Processing*, volume 43, pages 20–30. ACM, 2003.

[SG04] V. Surazhsky and C. Gotsman. High quality compatible triangulations. *Engineering with Computers*, 20(2) :147–156, 2004.

[Sha01] G. Sharma. Show-through cancellation in scans of duplex printed documents. *IEEE Transactions on Image Processing*, 10(5) :736–754, 2001.

[Sha08] A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6) :1539–1556, 2008.

[She02] J.R. Shewchuk. What is a Good Linear Element ? Interpolation, Conditioning, and Quality Measures. In *In 11th International Meshing Roundtable*, 2002.

[SJ07] C. Solnon and J.-M. Jolion. Generalized vs set median strings for histogram-based distances : algorithms and classification results in the image domain. In *5th IAPR-TC-15 workshop on Graph-based Representations in Pattern Recognition*, number 4538 in LNCS, pages 404–414 (poster). Springer, 2007.

[SLC04] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions : a local svm approach. In *ICPR*, volume 3, pages 32–36 Vol.3, September 2004.

[SM12] C. Sutton and A. McCallum. An introduction to conditional random fields. *Foundations and Trends in Machine Learning*, 4(4), 2012.

[SN09] P. Simari and D. Nowrouzezahrai. Multi-objective shape segmentation and labeling. *Computer Graphics Forum*, 28(5), 2009.

[Sol10] C. Solnon. Alldifferent-based filtering for subgraph isomorphism. *Artificial Intelligence*, 174(12-13) :850–864, 2010.

[SSHP97] J. Sauvola, T. Seppänen, S. Haapakoski, and M. Pietikäinen. Adaptive Document Binarization. In *International Conference on Document Analysis and Recognition*, volume 1, pages 147–152, 1997.

[SSS+09] L. Shapira, S. Shalom, A. Shamir, D. Cohen-Or, and H. Zhang. Contextual part analogies in 3d objects. *International Journal of Computer Vision*, 89(2-3) :309–326, 2009.

[SSSCO08] L. Shapira, S. Shalom, A. Shamir, and D. Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. In *The Visual Computer*, volume 24, pages 249–259, 2008.

[ST08] P. Sand and S. Teller. Particle video : Long-range motion estimation using point trajectories. *International Journal of Computer Vision*, 80(1) :72–91, 2008.

[SvKK+11] O. Sidi, O. van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or. Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering. *ACM Transactions on Graphics*, 30(6), 2011.

[SWK07] R. Schnabel, R. Wahl, and R. Klein. Efficient RANSAC for Point-Cloud Shape Detection. *Computer Graphics Forum*, 26(2) :214–226, June 2007.

[SZ03] J. Sivic and A. Zisserman. Video google : a text retrieval approach to object matching in videos. In *ICCV*, volume 2, pages 1470–1477, 2003.

[SZL92] W.J. Schroeder, J.A. Zarge, and W.E. Lorensen. Decimation of triangle meshes. In *SIGGRAPH*, volume 26, pages 65–70, 1992.

[Tau95] G. Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358, 1995.

[TB04] A. Tonazzini and L. Bedini. Independent component analysis for document restoration. *International Journal on Document Analysis and Recognition*, 7(1) :17–27, 2004.

[TBKL12] E. Tretyak, Olga Barinova, Pushmeet Kohli, and Viktor Lempitsky. Geometric image parsing of man-made environments. *International Journal of Computer Vision*, 97 :305–321, 2012.

[TBS06] A. Tonazzini, L. Bedini, and E. Salerno. A markov model for blind image separation by a mean-field em algorithm. *IEEE Transactions on Image Processing*, 15(2) :473–482, 2006.

[TCF10] R. Toldo, U. Castellani, and A. Fusiello. The bag of words approach for retrieval and categorization of 3d objects. *The Visual Computer*, 26(10) :1257–1268, 2010.

[TCS02] C.L. Tan, R. Cao, and P. Shen. Restoration of archival documents using a wavelet technique. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(10) :1399–1404, 2002.

[TCSB06] T.S.Caetano, T. Caelli, D. Schuurmans, and D.A.C. Barone. Graphical models and point pattern matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10) :1646–1663, 2006.

[TG05] A. Tonazzini and I. Gerace. Bayesian MRF-based blind source separation of convolutive mixtures of images. In *Proceedings of the 13th european signal processing conference*, 2005.

[THR07] G. W. Taylor, G. E. Hinton, and S. Roweis. Modeling human motion using binary latent variables. *Proc. NIPS*, 19, 2007.

[TKR08] L. Torresani, V. Kolmogorov, and C. Rother. Feature correspondence via graph matching : Models and global optimization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 2, pages 596–609, 2008.

[TLL+11] U. Tariq, K.H. Lin, Z. Li, X. Zhou, Z. Wang, V. Le, T.S. Huang, X. Lv, and T.X. Han. Emotion recognition from an ensemble of features. In *International Conference on Automatic Face & Gesture Recognition*, pages 872–877, 2011.

[TSB07] A. Tonazzini, E. Salerno, and L. Bedini. Fast correction of bleed-through distortion in grayscale documents by a blind source separation technique. *International Journal on Document Analysis and Recognition*, 10(1) :17–25, 2007.

[TVB03] A. Tonazzini, S. Vezzosi, and L. Bedini. Analysis and recognition of highly degraded printed characters. *International Journal on Document Analysis and Recognition*, 6(4) :236–247, 2003.

[TVD06] J. Tierny, J.-P. Vandeborre, and M. Daoudi. 3d Mesh Skeleton Extraction Using Topological and Geometrical Analyses. In *Pacific Conference on Computer Graphics and Applications*, pages 85–94, 2006.

[TVD07] J. Tierny, J.-P. Vandeborre, and M. Daoudi. Reeb chart unfolding based 3D shape signatures. In *Eurographics*, 2007. short paper.

[TWL+10] A.P. Ta, C. Wolf, G. Lavoué, A. Baskurt, and J-M. Jolion. Pairwise features for human action recognition. In *Proceedings of the International Conference on Pattern Recognition*, 2010.

[TWLB09] A.-P. Ta, C. Wolf, G. Lavoué, and A. Baskurt. 3D object detection and viewpoint selection in sketch images using local patch-based zernike moments. In *International workshop on content-based multimedia indexing*, pages 189–194, 2009.

[TWLB10] A.-P. Ta, C. Wolf, G. Lavoue, and A. Baskurt. Recognizing and localizing individual activities through graph matching. In *International Conference on Advanced Video and Signal-Based Surveillance (Best paper for track "recognition")*, 2010.

[Ull76] J.R. Ullmann. An algorithm for subgraph isomorphism. *Journal of the ACM*, 23(1) :31–42, 1976.

[VCP08] S. Valette, J.-M. Chassery, and R. Prost. Generic remeshing of 3D triangular meshes with metric-dependent discrete voronoi diagrams. *IEEE transactions on visualization and computer graphics*, 14(2) :369–381, 2008.

[Vid11] V. Vidal. *Développement de modèles graphiques probabilistes pour analyser et remailler les maillages triangulaires 2-variétés*. PhD thesis, INSA de Lyon, 2011.

[Vit67] A.J. Viterbi. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. *IEEE Transactions on Information Theory*, IT-13 :260–269, 1967.

[VJM+11] M.F. Valstar, B. Jiang, M. Mehu, M. Pantic, and K. Scherer. The first facial expression recognition and analysis challenge. In *International Conference on Automatic Face & Gesture Recognition*, pages 921–926, 2011.

[VWD11] V. Vidal, C. Wolf, and F. Dupont. Robust feature line extraction on cad triangular meshes,. In *Proceedings of the International Conference on Computer Graphics Theory and Applications*, 2011.

[VWD12a] V. Vidal, C. Wolf, and F. Dupont. Combinatorial mesh optimization. *The Visual Computer*, 28(5) :511–525, 2012.

[VWD12b] V. Vidal, C. Wolf, and F. Dupont. Mesh segmentation and global 3d model extraction. In *Poster at Symposium on Geometry Processing (SGP)*, 2012.

[VWLD09] V. Vidal, C. Wolf, G. Lavoué, and F. Dupont. Global triangular mesh regularization using conditional markov random fields. In *Poster at Symposium on Geometry Processing (acceptance rate 35%)*, 2009.

[WADP97] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder : real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 19(7) :780–785, 1997.

[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.

[Wei00] Y. Weiss. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12 :1–41, 2000.

[WF01] Y. Weiss and W.T. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEE Transactions on Information Theory*, 47(2) :736–744, 2001.

[WG10] C. Wolf and G. Gavin. Inference and parameter estimation on hierarchical belief networks for image segmentation. *Neurocomputing*, 43(4–6) :563–569, 2010.

[WHG08] T. Winkler, K. Hormann, and C. Gotsman. Mesh Massage : A Versatile Mesh Optimization Framework. *The Visual Computer*, 24(7-9) :775–785, 2008.

[WJ06] C. Wolf and J.-M. Jolion. Object count/Area Graphs for the Evaluation of Object Detection and Segmentation Algorithms. *International Journal on Document Analysis and Recognition*, 8(4) :280–296, 2006.

[WJ07] C. Wolf and J.-M. Jolion. Quality, quantity and generality in the evaluation of object detection algorithms. In *Proceedings of the Image Eval Conference*, 2007.

[WJ10] C. Wolf and J.M. Jolion. Integrating a discrete motion model into GMM based background subtraction. In *Proceedings of the International Conference on Pattern Recognition*, 2010.

[WJC02] C. Wolf, J.-M. Jolion, and F. Chassaing. Text Localization, Enhancement and Binarization in Multimedia Documents. In *Proceedings of the International Conference on Pattern Recognition*, volume 2, pages 1037–1040, 2002.

[WK05] J. Wu and L. Kobbelt. Structure recovery via hybrid variational surface approximation. *Computer Graphics Forum (Eurographics 2005 proceedings)*, 24(3) :277 – 284, 2005.

[WLL00] L. Wang, J. Liu, and S.Z. Li. Mrf parameter estimation by mcmc method. *Pattern Recognition*, 32(11) :1919–1925, 2000.

[WLW01] J.Z. Wang, J. Li, and G. Wiederhold. Simplicity : Semantics-sensitive integrated matching for picture libraries. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 23(9) :947–963, 2001.

[WML+12] C. Wolf, J. Mille, E. Lombardi, O. Celiktutan, M. Jiu, M. Baccouche, E. Dellandréa, C.-E. Bichot, C. Garcia, and B. Sankur. The liris human activities dataset and the icpr 2012 human activities recognition and localization competition. Technical Report LIRIS RR-2012-004, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2012.

[Wol06] C. Wolf. Document ink bleed-through removal with two hidden Markov random fields and a single observation field. Technical Report LIRIS RR-2006-019, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2006.

[Wol08] C. Wolf. Improving recto document side restoration with an estimation of the verso side from a single scanned page. In *Proceedings of the International Conference on Pattern Recognition*, 2008.

[Wol10] C. Wolf. Document ink bleed-through removal with two hidden markov random fields and a single observation field. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 32(3) :431–447, 2010.

[WS06] J. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *Proceeding of the conference on Computer Vision and Pattern Recognition(CVPR)*, volume 1, pages 37–44, 2006.

[WT11] C. Wolf and G. Taylor. Learning individual human activities from short binary shape sequences. Technical Report LIRIS RR-2011-018, Laboratoire d'Informatique en Images et Systèmes d'Information, INSA de Lyon, France, 2011.

[WTG08] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV*, 2008.

[WTSL08] Yu S. Wang, Chiew L. Tai, Olga Sorkine, and Tong Y. Lee. Optimized scale-and-stretch for image resizing. In *SIGGRAPH Asia*, pages 1–8, 2008.

[WXTL03] Q. Wang, T. Xia, C.L. Tan, and L. Li. Directional wavelet approach to remove document image interference. In *International Conference on Document Analysis and Recognition*, pages 736–740, 2003.

[XC08] D. Xu and S.-F. Chang. Video event recognition using kernel methods with multi-level temporal alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11) :1985–1997, 2008.

[XHW09] B. Xiaoa, E.R. Hancock, and R.C. Wilson. Graph characteristics from the heat kernel trace. *Pattern Recognition*, 42(2009) :2589–2606, 2009.

[YB11] S. Yang and B. Bhanu. Facial expression recognition using emotion avatar image. In *International Conference on Automatic Face & Gesture Recognition*, pages 866 –871, 2011.

[YFW05] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, 51(7) :2282–2312, 2005.

[YGZW07] W. Yue, Q. Guo, J. Zhang, and G. Wang. 3D triangular mesh optimization in geometry processing for CAD. In *Proceedings of the 2007 ACM symposium on Solid and physical modeling - SPM '07*, volume 1, pages 23–33, 2007.

[YJK+11] B. Yao, X. Jiang, A. Khosla, A.L. Lin, L.J. Guibas, and L. Fei-Fei. Action recognition by learning bases of action attributes and parts. In *International Conference on Computer Vision (ICCV)*, 2011.

[YLW06] D.M. Yan, Y. Liu, and W. Wang. Quadric surface extraction by variational shape approximation. In *Geometric Modeling and Processing*, pages 73–86, 2006.

[ZBV09] M. Zaslavskiy, F. Bach, and J.P. Vert. A path following algorithm for the graph matching problem. *IEEE Tr. on PAMI*, 31(12) :2227–2242, 2009.

[Zha92] J. Zhang. The mean field theory in em procedures for Markov random fields. *IEEE Transactions on Image Processing*, 40(10) :2570–2583, 1992.

[ZS08] R. Zass and A. Shashua. Probabilistic graph and hypergraph matching. In *CVPR*, 2008.

[ZSCP08] Y. Zeng, D. Samaras, W. Chen, and Q. Peng. Topology cuts : A novel min-cut/max-flow algorithm for topology preserving segmentation in n-d images. *Computer Vision and Image Understanding*, 112 :81—90, 2008.

[ZT10] Y. Zheng and C.L. Tai. Mesh decomposition with cross-boundary brushes. *Computer Graphics Forum*, 29(3) :527–535, 2010.

[ZvdH06] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7) :773–780, 2006.

[ZWW+10] Y. Zeng, C. Wang, Y. Wang, X. Gu, D. Samaras, and N. Paragios. Dense non-rigid surface registration using high-order graph matching. In *CVPR*, 2010.

[ZZWC12] J. Zhang, J. Zheng, C. Wu, and J. Cai. Variational mesh decomposition. *ACM Transactions on Graphics*, 31(3) :21 :1–21 :14, 2012.